



Cisco Aggregation Services Router 1004 (ASR1K)

Security Target

Version 2.0
16 August 2019



Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

© 2019 Cisco Systems, Inc. All rights reserved.

Table of Contents

Table of Contents	2
1 SECURITY TARGET INTRODUCTION.....	8
1.1 ST and TOE Reference	8
1.2 TOE Overview.....	8
1.2.1 TOE Product Type.....	9
1.2.2 Supported non-TOE Hardware/ Software/ Firmware.....	9
1.3 TOE DESCRIPTION	9
1.4 TOE Evaluated Configuration	10
1.5 Physical Scope of the TOE	12
1.6 Logical Scope of the TOE.....	13
1.6.1 Security Audit.....	13
1.6.2 Cryptographic Support.....	14
1.6.3 Identification and authentication	15
1.6.4 Security Management.....	16
1.6.5 Protection of the TSF.....	16
1.6.6 TOE Access.....	17
1.6.7 Trusted path/Channels.....	17
1.7 Excluded Functionality	17
2 Conformance Claims	18
2.1 Common Criteria Conformance Claim	18
2.2 Protection Profile Conformance	18
2.3 Protection Profile Conformance Claim Rationale.....	22
2.3.1 TOE Appropriateness.....	22
2.3.2 TOE Security Problem Definition Consistency	22
2.3.3 Statement of Security Requirements Consistency.....	22
3 SECURITY PROBLEM DEFINITION	23
3.1 Assumptions	23
3.2 Threats.....	24
3.3 Organizational Security Policies	26
4 SECURITY OBJECTIVES.....	27
4.1 Security Objectives for the TOE	27
4.2 Security Objectives for the Environment.....	27
5 SECURITY REQUIREMENTS.....	29
5.1 Conventions	29
5.2 TOE Security Functional Requirements	29
5.3 SFRs from NDcPP.....	31
5.3.1 Security audit (FAU).....	31

5.3.2	Cryptographic Support (FCS).....	34
5.3.3	Identification and authentication (FIA).....	38
5.3.4	Security management (FMT).....	40
5.3.5	Protection of the TSF (FPT).....	42
5.3.6	TOE Access (FTA).....	43
5.3.7	Trusted Path/Channels (FTP).....	43
5.4	TOE SFR Dependencies Rationale for SFRs Found in PP.....	44
5.5	Security Assurance Requirements.....	45
5.5.1	SAR Requirements.....	45
5.5.2	Security Assurance Requirements Rationale.....	45
5.6	Assurance Measures.....	46
6	<i>TOE Summary Specification.....</i>	47
6.1	TOE Security Functional Requirement Measures.....	47
7	<i>Annex A: Key Zeroization.....</i>	62
7.1	Key Zeroization.....	62
8	<i>Annex B: References.....</i>	65

List of Tables

TABLE 1 ACRONYMS	5
TABLE 2 TERMINOLOGY.....	6
TABLE 3 ST AND TOE IDENTIFICATION.....	8
TABLE 4 IT ENVIRONMENT COMPONENTS.....	9
TABLE 5 HARDWARE MODELS AND SPECIFICATIONS.....	13
TABLE 6 FIPS REFERENCES	14
TABLE 7 TOE PROVIDED CRYPTOGRAPHY	14
TABLE 8 EXCLUDED FUNCTIONALITY	17
TABLE 9 NIAP TECHNICAL DECISIONS (TD)	18
TABLE 10 PROTECTION PROFILES	22
TABLE 11 TOE ASSUMPTIONS.....	23
TABLE 12 THREATS.....	24
TABLE 13 ORGANIZATIONAL SECURITY POLICIES	26
TABLE 14 SECURITY OBJECTIVES FOR THE ENVIRONMENT	27
TABLE 15 SECURITY FUNCTIONAL REQUIREMENTS	29
TABLE 16 AUDITABLE EVENTS	31
TABLE 17 ASSURANCE MEASURES.....	45
TABLE 18 ASSURANCE MEASURES.....	46
TABLE 19 HOW TOE SFRs MEASURES.....	47
TABLE 20 TOE KEY ZEROIZATION	62
TABLE 21 REFERENCES	65

List of Figures

FIGURE 1 TOE EXAMPLE DEPLOYMENT	11
---------------------------------------	----

Acronyms

The following acronyms and abbreviations are common and may be used in this Security Target:

Table 1 Acronyms

Acronyms/Abbreviations	Definition
AAA	Administration, Authorization, and Accounting
AES	Advanced Encryption Standard
BRI	Basic Rate Interface
CC	Common Criteria for Information Technology Security Evaluation
CEM	Common Evaluation Methodology for Information Technology Security
CM	Configuration Management
CSU	Channel Service Unit
DHCP	Dynamic Host Configuration Protocol
DSU	Data Service Unit
EAL	Evaluation Assurance Level
EHWIC	Ethernet High-Speed WIC
ESP	Encapsulating Security Payload
ESPr	Embedded Services Processors
GE	Gigabit Ethernet port
HTTPS	Hyper-Text Transport Protocol Secure
IT	Information Technology
NDcPP	collaborative Protection Profile for Network Devices
OS	Operating System
PoE	Power over Ethernet
PP	Protection Profile
SA	Security Association
SFP	Small-form-factor pluggable port
SHS	Secure Hash Standard
ST	Security Target
TCP	Transport Control Protocol
TSC	TSF Scope of Control
TSF	TOE Security Function
TSP	TOE Security Policy
WAN	Wide Area Network
WIC	WAN Interface Card

Terminology

Table 2 Terminology

Term	Definition
Authorized Administrator	Any user which has been assigned to a privilege level that is permitted to perform all TSF-related functions.
Peer	Another router on the network that the TOE interfaces with.
Privilege level	Assigns a user specific management access to the TOE to run specific commands. The privilege levels are from 1-15 with 15 having full administrator access to the TOE similar to root access in UNIX or Administrator access on Windows. Privilege level 1 has the most limited access to the CLI. By default when a user logs in to the Cisco IOS-XE, they will be in user EXEC mode (level 1). From this mode, the administrator has access to some information about the TOE, such as the status of interfaces, and the administrator can view routes in the routing table. However, the administrator can't make any changes or view the running configuration file. The privilege levels are customizable so that an Authorized Administrator can also assign certain commands to certain privilege levels.
Remote VPN Gateway/Peer	A remote VPN Gateway/Peer is another network device that the TOE sets up a VPN connection with. This could be a VPN client or another router.
Role	An assigned role gives a user varying access to the management of the TOE. For the purposes of this evaluation the privilege level of user is synonymous with the assigned privilege level.
Security Administrator	Synonymous with Authorized Administrator for the purposes of this evaluation.
User	Any entity (human user or external IT entity) outside the TOE that interacts with the TOE.
Vty	vty is a term used by Cisco to describe a single terminal (whereas Terminal is more of a verb or general action term). For configuration purposes vty defines the line for remote access policies to the router.

DOCUMENT INTRODUCTION

Prepared By:
Cisco Systems, Inc.
170 West Tasman Dr.
San Jose, CA 95134

This document provides the basis for an evaluation of a specific Target of Evaluation (TOE), Cisco Aggregation Services Router 1004 (ASR1K). This Security Target (ST) defines a set of assumptions about the aspects of the environment, a list of threats that the product intends to counter, a set of security objectives, a set of security requirements, and the IT security functions provided by the TOE which meet the set of requirements.

1 SECURITY TARGET INTRODUCTION

The Security Target contains the following sections:

- Security Target Introduction [Section 1]
- Conformance Claims [Section 2]
- Security Problem Definition [Section 3]
- Security Objectives [Section 4]
- IT Security Requirements [Section 5]
- TOE Summary Specification [Section 6]

The structure and content of this ST comply with the requirements specified in the Common Criteria (CC), Part 1, Annex A, and Part 3, Chapter 11.

1.1 ST and TOE Reference

This section provides information needed to identify and control this ST and its TOE.

Table 3 ST and TOE Identification

Name	Description
ST Title	Cisco Aggregation Services Router 1004 (ASR1K) Security Target
ST Version	2.0
Publication Date	16 August 2019
Vendor and ST Author	Cisco Systems, Inc.
TOE Reference	Cisco Aggregation Services Router 1004 (ASR1K)
TOE Hardware Models	ASR1004
TOE Software Version	IOS-XE 16.12
Keywords	Router, Network Appliance, Data Protection, Authentication, Cryptography, Secure Administration, Network Device

1.2 TOE Overview

The Cisco Aggregation Services Router 1004 (herein after referred to as the ASR1K) is a purpose-built, routing platform. It performs analysis of incoming frames, makes forwarding decisions based on information contained in the frames, and forwards the frames toward the destination. It supports routing of traffic based on tables identifying available routes, conditions, distance, and costs to determine the best route for a given packet.

Cisco IOS-XE software is a Cisco-developed highly configurable proprietary operating system that provides for efficient and effective switching and routing. Although IOS performs many networking functions, this Security Target only addresses the functions that provide for the security of the TOE itself.

1.2.1 TOE Product Type

The ASR1K router aggregates multiple WAN connections and network services including encryption and traffic management, and forward them across WAN connections. The router contains both hardware and software redundancy in an industry-leading high-availability design. The ASR1K supports up to 8 shared port adapters (SPA) and comes with one route processor and one embedded services processor slot. This provides flexibility for future upgrades, enhancing control plane and data plane scalability.

1.2.2 Supported non-TOE Hardware/ Software/ Firmware

The TOE supports the following hardware, software, and firmware in its environment when the TOE is configured in its evaluated configuration:

Table 4 IT Environment Components

Component	Required	Usage/Purpose Description for TOE performance
RADIUS AAA Server	Yes	This includes any IT environment RADIUS AAA server that provides authentication services to TOE administrators.
Management Workstation with SSH Client	Yes	This includes any IT Environment Management workstation with a SSH client installed that is used by the TOE administrator to support TOE administration through SSH protected channels. Any SSH client that supports SSHv2 may be used.
Local Console	Yes	This includes any IT Environment Console that is directly connected to the TOE via the Serial Console Port and is used by the TOE administrator to support TOE administration.
Audit (syslog) Server	Yes	This includes any syslog server to which the TOE would transmit syslog messages. Also referred to as audit server in the ST.
Remote VPN Peer	Yes	This includes any VPN peer with which the TOE participates in VPN communications. Remote VPN Endpoints may be any device that supports IPsec VPN communications.
Certificate Authority (CA)	Yes	This includes any IT Environment Certification Authority (CA) on the TOE network. The CA can be used to provide the TOE with a valid certificate during certificate enrollment as well as validating a certificate.

1.3 TOE DESCRIPTION

This section provides an overview of the ASR1K Target of Evaluation (TOE). This section also defines the TOE components included in the evaluated configuration of the TOE. The ASR1K hardware models are included in the evaluation and consists of a number of components including:

- Chassis: The ASR1004 chassis is a 4-RU form factor. The chassis is the component of the TOE in which all other TOE components are housed.

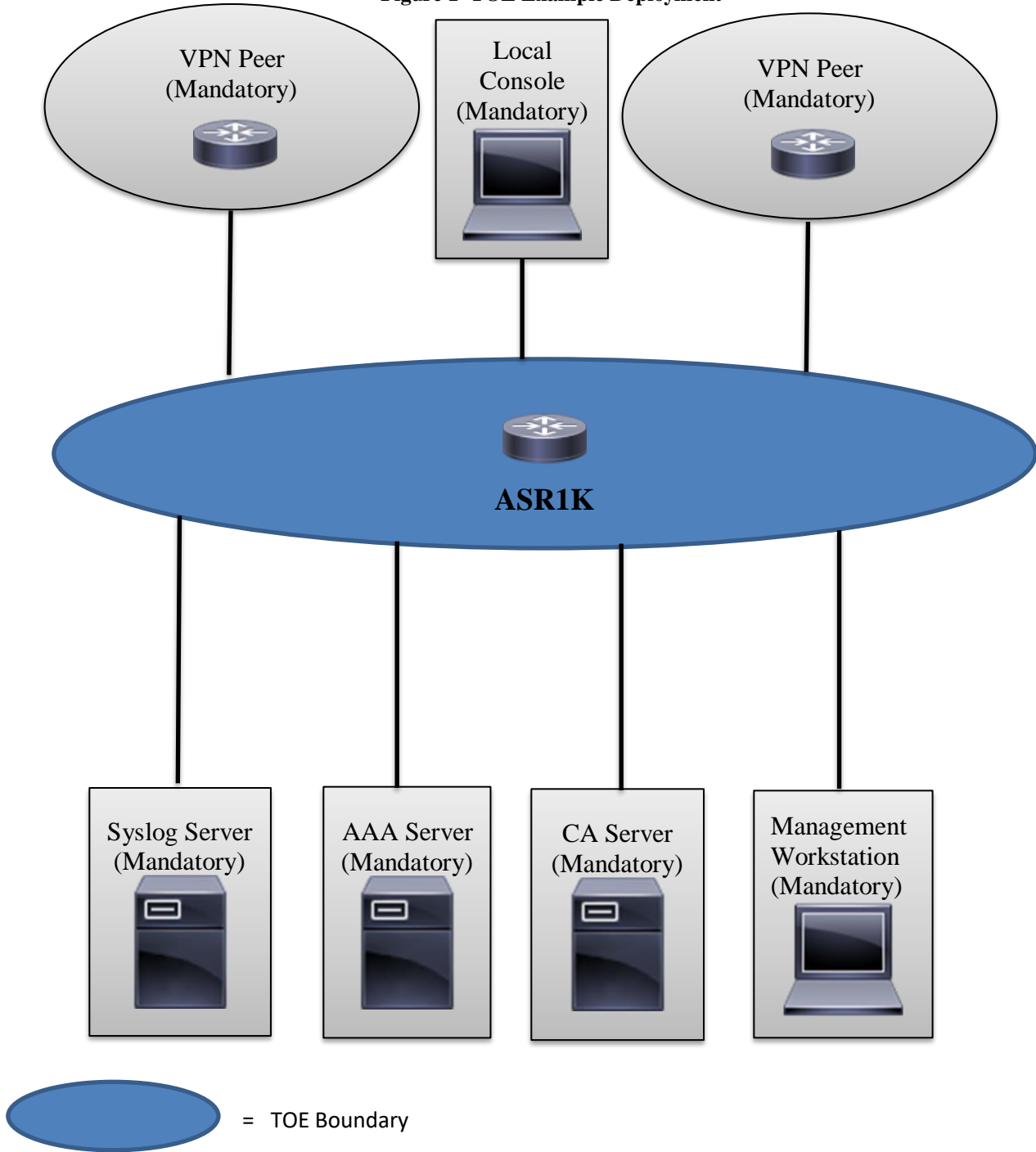
- Embedded Services Processor (ESP): The ASR1000-ESP20 and the ASR1000-ESP40 are responsible for the data-plane processing tasks, and all network traffic flows through them.
- Route Processor (RP): The ASR100-RP2 provides the advanced routing capabilities of the TOE. The RP also monitors and manages the other components in the ASR1K.
- Shared Port Adaptors (SPA): Used for connecting to networks. These SPAs interface with the TOE to provide the network interfaces that will be used to communicate on the network.

1.4 TOE Evaluated Configuration

The TOE consists of one or more physical devices as specified in section 1.5 below and includes the Cisco IOS-XE software. The TOE has two or more network interfaces and is connected to at least one internal and one external network. The Cisco IOS-XE configuration determines how packets are handled to and from the TOE's network interfaces. The router configuration will determine how traffic flows received on an interface will be handled. Typically, packet flows are passed through the internetworking device and forwarded to their configured destination.

The following figure provides a visual depiction of an example TOE deployment.

Figure 1 TOE Example Deployment



The previous figure includes the following:

- Examples of TOE Models
- The following are considered to be in the IT Environment:
 - VPN Peers
 - Management Workstation
 - Authentication Server
 - Syslog Server
 - CA Server
 - Local Console


1.5 Physical Scope of the TOE

The TOE is a hardware and software solution that makes up the router models as follows:

- Cisco ASR1004; ASR1000-ESP20 and ASR1000-ESP40; ASR1000-RP2

The network, on which they reside, is considered part of the environment. The software is pre-installed and is comprised of the Cisco IOS-XE software image Release 16.12. In addition, the software image is also downloadable from the Cisco web site. A login id and password is required to download the software image. The TOE is comprised of the following physical specifications as described in Table 5 below:

Table 5 Hardware Models and Specifications

Hardware	Picture	Size	Specifications
ASR1004 ASR1000-ESP20 ASR1000-ESP40 ASR1000-RP2		7 x 17.2 x 18.5 in.	<ul style="list-style-type: none"> • Shared Port Adapters: 8 • Ethernet Line cards: 2 • ESP slots: 1 • RP slots: 1 • ESP Bandwidth: 10 to 40 Gbps • Deafault memory: 8 GB DRAM • External USB flash memory: 1 GB support

1.6 Logical Scope of the TOE

The TOE is comprised of several security features. Each of the security features identified above consists of several security functionalities, as identified below.

1. Security Audit
2. Cryptographic Support
3. Identification and Authentication
4. Security Management
5. Protection of the TSF
6. TOE Access
7. Trusted Path/Channels

These features are described in more detail in the subsections below. In addition, the TOE implements all RFCs of the NDcPP v2.0e as necessary to satisfy testing/assurance measures prescribed therein.

1.6.1 Security Audit

The TOE provides extensive auditing capabilities. The TOE can audit events related to cryptographic functionality, identification and authentication, and administrative actions. The TOE generates an audit record for each auditable event. Each security relevant audit event has the date, timestamp, event description, and subject identity. The administrator configures auditable events, performs back-up operations and manages audit data storage. The TOE provides the administrator with a circular audit trail or a configurable audit trail threshold to track the storage capacity of the audit trail. Audit logs are backed up over an encrypted channel to an external audit server.

1.6.2 Cryptographic Support

The TOE provides cryptography in support of other TOE security functionality. All the algorithms claimed have CAVP certificates (Operation Environment - Intel Xeon 5200) (see Table 6 for certificate references).

Table 6 FIPS References

Algorithm	Description	Supported Mode	Module	CAVP Cert. #	SFR
AES	Used for symmetric encryption/decryption	CBC (128, 192 and 256)	IC2M	C 462	FCS_COP.1/DataEncryption
SHS (SHA-1, SHA-256 and SHA-512)	Cryptographic hashing services	Byte Oriented	IC2M	C 462	FCS_COP.1/Hash
HMAC (HMAC-SHA-1)	Keyed hashing services and software integrity test	Byte Oriented	IC2M	C 462	FCS_COP.1/KeyedHash
DRBG	Deterministic random bit generation services in accordance with ISO/IEC 18031:2011	CTR_DRBG (AES 256)	IC2M	C 462	FCS_RBG_EXT.1
RSA	Signature Verification and key transport	PKCS#1 v.1.5, 2048 bit key, FIPS 186-4 Key Gen	IC2M	C 462	FCS_CKM.1 FCS_COP.1/SigGen
ECDSA	Cryptographic Signature services	FIPS 186-4, Digital Signature Standard (DSS)	IC2M	C 462	FCS_CKM.1 FCS_COP.1/SigGen
CVL-KAS-ECC	Key Agreement	NIST Special Publication 800-56A	IC2M	C 462	FCS_CKM.2

The TOE provides cryptography in support of VPN connections and remote administrative management via SSHv2 and IPsec to secure the transmission of audit records to the remote syslog server. In addition, IPsec is used to secure the session between the TOE and the authentication servers.

The cryptographic services provided by the TOE are described in Table 7 below.

Table 7 TOE Provided Cryptography

Cryptographic Method	Use within the TOE
Internet Key Exchange	Used to establish initial IPsec session.

Cryptographic Method	Use within the TOE
Secure Shell Establishment	Used to establish initial SSH session.
RSA Signature Services	Used in IPsec session establishment. Used in SSH session establishment. X.509 certificate signing
SP 800-90 RBG	Used in IPsec session establishment. Used in SSH session establishment.
SHS	Used to provide IPsec traffic integrity verification Used to provide SSH traffic integrity verification Used for keyed-hash message authentication
AES	Used to encrypt IPsec session traffic. Used to encrypt SSH session traffic.
RSA	Used in IKE protocols peer authentication Used to provide cryptographic signature services
ECC	Used to provide cryptographic signature services Used in Cryptographic Key Generation and Key Establishment
FFC DH	Used as the Key exchange method for SSH and IPsec
ECC DH	Used as the Key exchange method for IPsec

1.6.3 Identification and authentication

The TOE performs two types of authentication: device-level authentication of the remote device (VPN peers) and user authentication for the Authorized Administrator of the TOE. Device-level authentication allows the TOE to establish a secure channel with a trusted peer. The secure channel is established only after each device authenticates the other. Device-level authentication is performed via IKE/IPsec mutual authentication. The TOE supports use of IKEv1 (ISAKMP) and IKEv2 pre-shared keys for authentication of IPsec tunnels. The IKE phase authentication for the IPsec communication channel between the TOE and authentication server and between the TOE and syslog server is considered part of the Identification and Authentication security functionality of the TOE.

The TOE provides authentication services for administrative users to connect to the TOE's secure CLI administrator interface. The TOE requires Authorized Administrators to authenticate prior to being granted access to any of the management functionality. The TOE can be configured to require a minimum password length of 15 characters. The TOE provides administrator authentication against a local user database. Password-based authentication can be

performed on the serial console or SSH interfaces. The SSHv2 interface also supports authentication using SSH keys. The TOE supports the use of a RADIUS AAA server (part of the IT Environment) for authentication of administrative users attempting to connect to the TOE's CLI.

The TOE provides an automatic lockout when a user attempts to authenticate and enters invalid information. After a defined number of authentication attempts fail exceeding the configured allowable attempts, the user is locked out until an authorized administrator can enable the user account.

The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections.

1.6.4 Security Management

The TOE provides secure administrative services for management of general TOE configuration and the security functionality provided by the TOE. All TOE administration occurs either through a secure SSHv2 session or via a local console connection. The TOE provides the ability to securely manage:

- Administration of the TOE locally and remotely;
- All TOE administrative users;
- All identification and authentication;
- All audit functionality of the TOE;
- All TOE cryptographic functionality;
- The timestamps maintained by the TOE;
- Update to the TOE and verification of the updates.

The TOE supports two separate administrator roles: non-privileged administrator and privileged administrator. Only the privileged administrator can perform the above security relevant management functions. Management of the TSF data is restricted to Security Administrators. The ability to enable, disable, determine and modify the behavior of all of the security functions of the TOE is restricted to authorized administrators.

Administrators can create configurable login banners to be displayed at time of login, and can also define an inactivity timeout for each admin interface to terminate sessions after a set period of inactivity.

1.6.5 Protection of the TSF

The TOE protects against interference and tampering by untrusted subjects by implementing identification, authentication, and access controls to limit configuration to Authorized Administrators. The TOE prevents reading of cryptographic keys and passwords. Additionally, Cisco IOS-XE is not a general-purpose operating system and access to Cisco IOS-XE memory space is restricted to only Cisco IOS-XE functions.

The TOE internally maintains the date and time. This date and time is used as the timestamp that is applied to audit records generated by the TOE. Administrators can update the TOE's clock

manually. Finally, the TOE performs testing to verify correct operation of the router itself and that of the cryptographic module.

The TOE is able to verify any software updates prior to the software updates being installed on the TOE to avoid the installation of unauthorized software.

Whenever a failure occurs within the TOE that results in the TOE ceasing operation, the TOE securely disables its interfaces to prevent the unintentional flow of any information to or from the TOE and reloads.

1.6.6 TOE Access

The TOE can terminate inactive sessions after an Authorized Administrator configurable time-period. Once a session has been terminated the TOE requires the user to re-authenticate to establish a new session. Sessions can also be terminated if an Authorized Administrator enters the “exit” command.

The TOE can also display a Security Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE.

1.6.7 Trusted path/Channels

The TOE allows trusted paths to be established to itself from remote administrators over SSHv2, and initiates outbound IPsec tunnels to transmit audit messages to remote syslog servers. In addition, IPsec is used to secure the session between the TOE and the authentication servers. The TOE can also establish trusted paths of peer-to-peer IPsec sessions. The peer-to-peer IPsec sessions can be used for securing the communications between the TOE and authentication server/syslog server.

1.7 Excluded Functionality

The following functionality is excluded from the evaluation.

Table 8 Excluded Functionality

Excluded Functionality	Exclusion Rationale
Non-FIPS 140-2 mode of operation	This mode of operation includes non-FIPS allowed operations.

These services will be disabled by configuration settings as described in the Guidance documents (AGD). The exclusion of this functionality does not affect compliance to the NDcPP v2.0e.

2 CONFORMANCE CLAIMS

2.1 Common Criteria Conformance Claim

The TOE and ST are compliant with the Common Criteria (CC) Version 3.1, Revision 4, dated: September 2012. For a listing of Assurance Requirements claimed see section 5.5.

The TOE and ST are CC Part 2 extended and CC Part 3 conformant.

2.2 Protection Profile Conformance

The TOE and ST are conformant with the Protection Profiles as listed in Table 10 Protection Profiles below. The following NIAP Technical Decisions (TD) have also been applied to the claims in this document. Each posted TD was reviewed and considered based on the TOE product type, the PP claims and the security functional requirements claimed in this document.

Table 9 NIAP Technical Decisions (TD)

TD Identifier	TD Name	Protection Profiles	References	Publication Date	Applicable?
TD0412	NIT Technical Decision for FCS_SSHS_EXT.1.5 SFR and AA discrepancy	CPP_FW_V 2.0E, CPP_ND_V 2.0, CPP_ND_V 2.1	FCS_SSHS_EXT.1.5, ND SD V2.0e, ND SD V2.1	2019.03.22	Yes
TD0411	NIT Technical Decision for FCS_SSHC_EXT.1.5, Test 1 - Server and client side seem to be confused	CPP_FW_V 2.0E, CPP_ND_V 2.0E, CPP_ND_V 2.1	FCS_SSHC_EXT.1.5, ND SD V2.0E, ND SD V2.1	2019.03.22	No, SFR is not claimed
TD0410	NIT technical decision for Redundant assurance activities associated with FAU_GEN.1	CPP_ND_V 1.0, CPP_ND_V 2.0E, CPP_ND_V 2.1	FAU_GEN.1, ND SD V1.0, ND SD V2.0e, ND SD V2.1	2019.03.22	Yes
TD0409	NIT decision for Applicability of FIA_AFL.1 to key-based SSH authentication	CPP_ND_V 2.0E, CPP_ND_V 2.1	FIA_AFL.1, ND SD v2.0e, ND SD v2.1	2019.03.22	Yes
TD0408	NIT Technical Decision for local vs. remote administrator accounts	CPP_FW_V 2.0E, CPP_ND_V 2.0E, CPP_ND_V 2.1	FIA_AFL.1, FIA_UAU_EXT.2, FMT_SMF.1	2019.03.22	Yes
TD0407	NIT Technical Decision for handling Certification of Cloud Deployments	CPP_ND_V 2.0E, CPP_ND_V 2.1		2019.03.22	No, TOE is not a cloud

					deployment
TD0402	NIT Technical Decision for RSA-based FCS_CKM.2 Selection	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FCS_CKM.2, ND SD V2.0E	2019.02.24	Yes
TD0401	NIT Technical Decision for Reliance on external servers to meet SFRs	CPP_ND_V 2.0E	FTP_ITC.1	2019.02.24	Yes
TD0400	NIT Technical Decision for FCS_CKM.2 and elliptic curve-based key establishment	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FCS_CKM.1, FCS_CKM.2	2019.02.24	Yes
TD0399	NIT Technical Decision for Manual installation of CRL (FIA_X509_EXT.2)	CPP_ND_V 2.0E	FIA_X509_EXT.2, ND SD V2.0E	2019.02.24	Yes
TD0398	NIT Technical Decision for FCS_SSH*EXT.1.1 RFCs for AES-CTR	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FCS_SSHC_EXT.1.1, FCS_SSHS_EXT.1.1	2019.02.24	Yes
TD0397	NIT Technical Decision for Fixing AES-CTR Mode Tests	CPP_ND_V 2.0E	FCS_COP.1/DataEncryption, ND SD V2.0E	2019.02.24	Yes
TD0396	NIT Technical Decision for FCS_TLSC_EXT.1.1, Test 2	CPP_ND_V 2.0E	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, ND SD V2.0E	2019.02.24	No – TLS not claimed
TD0395	NIT Technical Decision for Different Handling of TLS1.1 and TLS1.2	CPP_ND_V 2.0E	FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5, ND SD V2.0E	2019.02.24	No – TLS not claimed
TD0394	NIT Technical Decision for Audit of Management Activities related to Cryptographic Keys	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FAU_GEN.1, ND SD v2.0E	2019.02.24	Yes
TD0343	NIT Technical Decision for Updating FCS_IPSEC_EXT.1.14 Tests	CPP_FW_V 2.0E, CPP_ND_V 2.0E	ND SD V2.0, FCS_IPSEC_EXT.1.14	2018.09.20	Yes
TD0342	NIT Technical Decision for TLS and DTLS Server Tests	CPP_ND_V 2.0E	ND SD V2.0, FCS_DTLSS_EXT.1, FCS_DTLSS_EXT.2, FCS_TLSS_EXT.1, FCS_TLSS_EXT.2	2018.08.02	No – TLS not claimed
TD0341	NIT Technical Decision for TLS wildcard checking	CPP_ND_V 2.0E	ND SD V2.0, FCS_TLSC_EXT.1.2, FCS_TLSC_EXT.2.2, FCS_DTLSC_EXT.1.2, FCS_DTLSC_EXT.2.2,	2018.08.02	No – TLS not claimed
TD0340	NIT Technical Decision for Handling of the basicConstraints extension in CA and leaf certificates	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FIA_X509_EXT.1.1	2018.08.02	Yes
TD0339	NIT Technical Decision for Making password-based	CPP_FW_V 2.0E,	ND SD V2.0, FCS_SSHS_EXT.1.2	2018.08.02	Yes

	authentication optional in FCS_SSHS_EXT.1.2	CPP_ND_V 2.0E			
TD0338	NIT Technical Decision for Access Banner Verification	CPP_ND_V 2.0E	ND SD V2.0, FTA_TAB.1	2018.08.02	Yes
TD0337	NIT Technical Decision for Selections in FCS_SSH*_EXT.1.6	CPP_FW_V 2.0E, CPP_ND_V 2.0E	ND SD V2.0, FCS_SSHC_EXT.1, FCS_SSHS_EXT.1	2018.08.02	Yes
TD0336	NIT Technical Decision for Audit requirements for FCS_SSH*_EXT.1.8	CPP_ND_V 2.0E	ND SD V2.0, FCS_SSHC_EXT.1.8, FCS_SSHS_EXT.1.8	2018.08.02	Yes
TD0335	NIT Technical Decision for FCS_DTLS Mandatory Cipher Suites	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FCS_DTLSC_EXT.1.1, FCS_DTLSC_EXT.2.1, FCS_DTLSS_EXT.1.1, FCS_DTLSS_EXT.2.1,	2018.08.01	No - TLS not claimed
TD0334	NIT Technical Decision for Testing SSH when password-based authentication is not supported	CPP_ND_V 2.0E	ND SD V2.0, FCS_SSHC_EXT.1.9	2018.08.01	No – FCS_SS HC not claimed
TD0333	NIT Technical Decision for Applicability of FIA_X509_EXT.3	CPP_FW_V 2.0E, CPP_ND_V 2.0E	ND SD V2.0, FIA_X509_EXT	2018.08.01	Yes
TD0324	NIT Technical Decision for Correction of section numbers in SD Table 1	CPP_ND_V 2.0E	Table 1	2018.05.18	Yes
TD0323	NIT Technical Decision for DTLS server testing - Empty Certificate Authorities list	CPP_ND_V 2.0E	ND SD V2.0, FCS_DTLSS_EXT.2.7, FCS_DTLSS_EXT.2.8	2018.05.18	No - TLS not claimed
TD0322	NIT Technical Decision for TLS server testing - Empty Certificate Authorities list	CPP_ND_V 2.0E	ND SD V.1.0, ND SD V2.0, FCS_TLSS_EXT.2.4, FCS_TLSS_EXT.2.5	2018.05.18	No - TLS not claimed
TD0321	Protection of NTP communications	CPP_FW_V 2.0E, CPP_ND_V 2.0E	FTP_ITC.1, FPT_STM_EXT.1	2018.05.21	Yes
TD0291	NIT technical decision for DH14 and FCS_CKM.1	CPP_FW_V 1.0, CPP_FW_v 2.0, CPP_FW_V 2.0E, CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	FCS_CKM.1.1, ND SD V1.0, ND SD V2.0	2018.02.03	Yes
TD0290	NIT technical decision for physical interruption of trusted path/channel.	CPP_ND_V 1.0, CPP_ND_V 2.0,	FTP_ITC.1, FTP_TRP.1, FPT_ITT.1, ND SD V1.0, ND SD V2.0	2018.02.03	Yes

		CPP_ND_V 2.0E			
TD0289	NIT technical decision for FCS_TLSC_EXT.x.1 Test 5e	CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	FCS_TLSC_EXT.1.1, FCS_TLSC_EXT.2.1, FCS_DTLSC_EXT.1.1 (only ND SD V2.0), FCS_DTLSC_EXT.2.1 (only ND SD V2.0)	2018.02.03	No - TLS not claimed
TD0281	NIT Technical Decision for Testing both thresholds for SSH rekey	CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	FCS_SSHC_EXT.1.8, FCS_SSHS_EXT.1.8, ND SD V1.0, ND SD V2.0	2018.01.05	Yes
TD0260	NIT Technical Decision for Typo in FCS_SSHS_EXT.1.4	CPP_FW_v 2.0, CPP_FW_V 2.0E, CPP_ND_V 2.0, CPP_ND_V 2.0E	FCS_SSHS_EXT.1.4	2017.11.13	Yes
TD0259	NIT Technical Decision for Support for X509 ssh rsa authentication IAW RFC 6187	CPP_FW_v 2.0, CPP_FW_V 2.0E, CPP_ND_V 2.0, CPP_ND_V 2.0E	FCS_SSHC_EXT.1.5/F CS_SSHS_EXT.1.5	2017.11.13	Yes
TD0257	NIT Technical Decision for Updating FCS_DTLSC_EXT.x.2/FCS_TLSC_EXT.x.2 Tests 1-4	CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	ND SD V1.0, ND SD V2.0, FCS_DTLSC_EXT.1.2/ FCS_DTLSC_EXT.2.2 Tests 1-4 (ND SD V2.0), FCS_TLSC_EXT.1.2/F CS_TLSC_EXT.2.2, Tests 1-4 (ND SD V1.0, ND SD V2.0)	2017.11.13	No - TLS not claimed
TD0256	NIT Technical Decision for Handling of TLS connections with and without mutual authentication	CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	ND SD V1.0, ND SD V2.0, FCS_DTLSC_EXT.2.5 (ND SD V2.0), FCS_TLSC_EXT.2 (ND SD V1.0, ND SD V2.0)	2017.11.13	No - TLS not claimed
TD0228	NIT Technical Decision for CA certificates - basicConstraints validation	CPP_FW_V 1.0, CPP_ND_V 1.0, CPP_ND_V 2.0, CPP_ND_V 2.0E	ND SD V1.0, ND SD V2.0, FIA_X509_EXT.1.2	2018.06.15	Yes

Table 10 Protection Profiles

Protection Profile	Version	Date
collaborative Protection Profile for Network Devices (NDcPP) Version 2.0 + Errata 20180314	2.0e	March 14, 2018

2.3 Protection Profile Conformance Claim Rationale

2.3.1 TOE Appropriateness

The TOE provides all of the functionality at a level of security commensurate with that identified in the U.S. Government Protection Profile and extended package:

- collaborative Protection Profile for Network Devices (NDcPP) + Errata 20180314, Version 2.0e

2.3.2 TOE Security Problem Definition Consistency

The Assumptions, Threats, and Organizational Security Policies included in the Security Target represent the Assumptions, Threats, and Organizational Security Policies specified in the collaborative Protection Profile for Network Devices (NDcPP) + Errata 20180314, Version 2.0e for which conformance is claimed verbatim. All concepts covered in the Protection Profile Security Problem Definition are included in the Security Target Statement of Security Objectives Consistency.

The Security Objectives included in the Security Target represent the Security Objectives specified in the NDcPP + Errata 20180314, Version 2.0e for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Objectives are included in the Security Target.

2.3.3 Statement of Security Requirements Consistency

The Security Functional Requirements included in the Security Target represent the Security Functional Requirements specified in the NDcPP v2.0e for which conformance is claimed verbatim. All concepts covered in the Protection Profile's Statement of Security Requirements are included in this Security Target. Additionally, the Security Assurance Requirements included in this Security Target are identical to the Security Assurance Requirements included in the NDcPP + Errata 20180314, Version 2.0e.

3 SECURITY PROBLEM DEFINITION

This chapter identifies the following:

- Significant assumptions about the TOE’s operational environment.
- IT related threats to the organization countered by the TOE.
- Environmental threats requiring controls to provide sufficient protection.
- Organizational security policies for the TOE as appropriate.

This document identifies assumptions as A.assumption with “assumption” specifying a unique name. Threats are identified as T.threat with “threat” specifying a unique name. Organizational Security Policies (OSPs) are identified as P.osp with “osp” specifying a unique name.

3.1 Assumptions

The specific conditions listed in the following subsections are assumed to exist in the TOE’s environment. These assumptions include both practical realities in the development of the TOE security requirements and the essential environmental conditions on the use of the TOE.

Table 11 TOE Assumptions

Assumption	Assumption Definition
A.PHYSICAL_PROTECTION	The network device is assumed to be physically protected in its operational environment and not subject to physical attacks that compromise the security and/or interfere with the device’s physical interconnections and correct operation. This protection is assumed to be sufficient to protect the device and the data it contains. As a result, the cPP will not include any requirements on physical tamper protection or other physical attack mitigations. The cPP will not expect the product to defend against physical access to the device that allows unauthorized entities to extract data, bypass other controls, or otherwise manipulate the device.
A.LIMITED_FUNCTIONALITY	The device is assumed to provide networking functionality as its core function and not provide functionality/ services that could be deemed as general purpose computing. For example the device should not provide computing platform for general purpose applications (unrelated to networking functionality).
A.NO_THRU_TRAFFIC_PROTECTION	A standard/generic network device does not provide any assurance regarding the protection of traffic that traverses it. The intent is for the network device to protect data that originates on or is destined to the device itself, to include administrative data and audit data. Traffic that is traversing the network device, destined for another network entity, is not covered by the ND cPP. It is assumed that this protection will be covered by cPPs for particular types of network devices (e.g. firewall).
A.TRUSTED_ADMINISTRATOR	The Security Administrator(s) for the network device are assumed to be trusted and to act in the best interest of security for the organization. This includes being appropriately trained, following policy, and adhering to guidance documentation. Administrators are trusted to ensure passwords/credentials have sufficient strength and

Assumption	Assumption Definition
	entropy and to lack malicious intent when administering the device. The network device is not expected to be capable of defending against a malicious administrator that actively works to bypass or compromise the security of the device.
A.REGULAR_UPDATES	The network device firmware and software is assumed to be updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
A.ADMIN_CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the network device are protected by the platform on which they reside.
A.RESIDUAL_INFORMATION	The Administrator must ensure that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

3.2 Threats

The following table lists the threats addressed by the TOE and the IT Environment. The assumed level of expertise of the attacker for all the threats identified below is Enhanced-Basic.

Table 12 Threats

Threat	Threat Definition
T.UNAUTHORIZED_ADMINISTRATOR_ACCESS	Threat agents may attempt to gain administrator access to the network device by nefarious means such as masquerading as an administrator to the device, masquerading as the device to an administrator, replaying an administrative session (in its entirety, or selected portions), or performing man-in-the-middle attacks, which would provide access to the administrative session, or sessions between network devices. Successfully gaining administrator access allows malicious actions that compromise the security functionality of the device and the network on which it resides.
T.WEAK_CRYPTOGRAPHY	Threat agents may exploit weak cryptographic algorithms or perform a cryptographic exhaust against the key space. Poorly chosen encryption algorithms, modes, and key sizes will allow attackers to compromise the algorithms, or brute force exhaust the key space and give them unauthorized access allowing them to read, manipulate and/or control the traffic with minimal effort.

Threat	Threat Definition
T.UNTRUSTED_COMMUNICATION_CHANNELS	Threat agents may attempt to target network devices that do not use standardized secure tunneling protocols to protect the critical network traffic. Attackers may take advantage of poorly designed protocols or poor key management to successfully perform man-in-the-middle attacks, replay attacks, etc. Successful attacks will result in loss of confidentiality and integrity of the critical network traffic, and potentially could lead to a compromise of the network device itself.
T.WEAK_AUTHENTICATION_ENDPOINTS	Threat agents may take advantage of secure protocols that use weak methods to authenticate the endpoints – e.g., shared password that is guessable or transported as plaintext. The consequences are the same as a poorly designed protocol, the attacker could masquerade as the administrator or another device, and the attacker could insert themselves into the network stream and perform a man-in-the-middle attack. The result is the critical network traffic is exposed and there could be a loss of confidentiality and integrity, and potentially the network device itself could be compromised.
T.UPDATE_COMPROMISE	Threat agents may attempt to provide a compromised update of the software or firmware which undermines the security functionality of the device. Non-validated updates or updates validated using non-secure or weak cryptography leave the update firmware vulnerable to surreptitious alteration.
T.UNDETECTED_ACTIVITY	Threat agents may attempt to access, change, and/or modify the security functionality of the network device without administrator awareness. This could result in the attacker finding an avenue (e.g., misconfiguration, flaw in the product) to compromise the device and the administrator would have no knowledge that the device has been compromised.
T.SECURITY_FUNCTIONALITY_COMPROMISE	Threat agents may compromise credentials and device data enabling continued access to the network device and its critical data. The compromise of credentials include replacing existing credentials with an attacker's credentials, modifying existing credentials, or obtaining the administrator or device credentials for use by the attacker.
T.PASSWORD_CRACKING	Threat agents may be able to take advantage of weak administrative passwords to gain privileged access to the device. Having privileged access to the device provides the attacker unfettered access to the network traffic, and may allow them to take advantage of any trust relationships with other network devices.

Threat	Threat Definition
T.SECURITY_FUNCTIONALITY_FAILURE	An external, unauthorized entity could make use of failed or compromised security functionality and might therefore subsequently use or abuse security functions without prior authentication to access, change or modify device data, critical network traffic or security functionality of the device.

3.3 Organizational Security Policies

The following table lists the Organizational Security Policies imposed by an organization to address its security needs.

Table 13 Organizational Security Policies

Policy Name	Policy Definition
P.ACCESS_BANNER	The TOE shall display an initial banner describing restrictions of use, legal agreements, or any other appropriate information to which users consent by accessing the TOE.

4 SECURITY OBJECTIVES

This Chapter identifies the security objectives of the TOE and the IT Environment. The security objectives identify the responsibilities of the TOE and the TOE's IT environment in meeting the security needs.

- This document identifies objectives of the TOE as O.objective with objective specifying a unique name. Objectives that apply to the IT environment are designated as OE.objective with objective specifying a unique name.

4.1 Security Objectives for the TOE

The collaborative Protection Profile for Network Devices v2.0 + Errata 20180314 does not define any security objectives for the TOE.

4.2 Security Objectives for the Environment

All of the assumptions stated in section 3.1 are considered to be security objectives for the environment. The following are the Protection Profile non-IT security objectives, which, in addition to those assumptions, are to be satisfied without imposing technical requirements on the TOE. That is, they will not require the implementation of functions in the TOE hardware and/or software. Thus, they will be satisfied largely through application of procedural or administrative measures.

Table 14 Security Objectives for the Environment

Environment Security Objective	IT Environment Security Objective Definition
OE.PHYSICAL	Physical security, commensurate with the value of the TOE and the data it contains, is provided by the environment.
OE.NO_GENERAL_PURPOSE	There are no general-purpose computing capabilities (e.g., compilers or user applications) available on the TOE, other than those services necessary for the operation, administration and support of the TOE.
OE.NO_THRU_TRAFFIC_PROTECTION	The TOE does not provide any protection of traffic that traverses it. It is assumed that protection of this traffic will be covered by other security and assurance measures in the operational environment.
OE.TRUSTED_ADMIN	Security Administrators are trusted to follow and apply all guidance documentation in a trusted manner.
OE.UPDATES	The TOE firmware and software is updated by an administrator on a regular basis in response to the release of product updates due to known vulnerabilities.
OE.ADMIN_CREDENTIALS_SECURE	The administrator's credentials (private key) used to access the TOE must be protected on any other platform on which they reside.

Environment Security Objective	IT Environment Security Objective Definition
OE.RESIDUAL_INFORMATION	The Security Administrator ensures that there is no unauthorized access possible for sensitive residual information (e.g. cryptographic keys, keying material, PINs, passwords etc.) on networking equipment when the equipment is discarded or removed from its operational environment.

5 SECURITY REQUIREMENTS

This section identifies the Security Functional Requirements for the TOE. The Security Functional Requirements included in this section are derived from Part 2 of the *Common Criteria for Information Technology Security Evaluation, Version 3.1, Revision 4, dated: September 2012* and all international interpretations.

5.1 Conventions

The CC defines operations on Security Functional Requirements: assignments, selections, assignments within selections and refinements. This document uses the following font conventions to identify the operations defined by the CC:

- Assignment: Indicated with *italicized* text;
- Assignment completed within a selection in the cPP: the completed assignment text is indicated with *italicized and underlined text*
- Refinement: Indicated with **bold** text;
- Selection: Indicated with underlined text;
- Iteration: Indicated by appending the iteration number in parenthesis, e.g., (1), (2), (3).
- Where operations were completed in the NDcPP itself, the formatting used in the NDcPP has been retained.

5.2 TOE Security Functional Requirements

This section identifies the Security Functional Requirements for the TOE. The TOE Security Functional Requirements that appear in the following table are described in more detail in the following subsections.

Table 15 Security Functional Requirements

Class Name	Component Identification	Component Name
FAU: Security audit	FAU_GEN.1	Audit Data Generation
	FAU_GEN.2	User identity association
	FAU_STG_EXT.1	Protected Audit Event Storage
FCS: Cryptographic support	FCS_CKM.1	Cryptographic Key Generation (Refined)
	FCS_CKM.2	Cryptographic Key Establishment (Refined)
	FCS_CKM.4	Cryptographic Key Destruction
	FCS_COP.1/DataEncryption	Cryptographic Operation (AES Data Encryption)
	FCS_COP.1/SigGen	Cryptographic Operation (Signature Verification)
	FCS_COP.1/Hash	Cryptographic Operation (Hash Algorithm)
	FCS_COP.1/KeyedHash	Cryptographic Operation (Keyed Hash Algorithm)

Class Name	Component Identification	Component Name
	FCS_IPSEC_EXT.1	IPsec Protocol
	FCS_SSHS_EXT.1	SSH Server Protocol
	FCS_RBG_EXT.1	Random Bit Generation
FIA: Identification and authentication	FIA_AFL.1	Authentication Failure Management (Refined)
	FIA_PMG_EXT.1	Password Management
	FIA_UIA_EXT.1	User Identification and Authentication
	FIA_UAU_EXT.2	Password-based Authentication Mechanism
	FIA_UAU.7	Protected Authentication Feedback
	FIA_X509_EXT.1/Rev	X.509 Certificate Validation
	FIA_X509_EXT.2	X.509 Certificate Authentication
	FIA_X509_EXT.3	X.509 Certificate Requests
FMT: Security management	FMT_MOF.1/ManualUpdate	Trusted Update - Management of security functions behaviour
	FMT_MTD.1/CoreData	Management of TSF Data
	FMT_SMF.1	Specification of Management Functions
	FMT_SMR.2	Restrictions on security roles
FPT: Protection of the TSF	FPT_SKP_EXT.1	Protection of TSF Data (for reading of all symmetric keys)
	FPT_APW_EXT.1	Protection of Administrator Passwords
	FPT_TST_EXT.1	TSF Testing (Extended)
	FPT_TUD_EXT.1	Extended: Trusted Update
	FPT_STM_EXT.1	Reliable Time Stamps
FTA: TOE Access	FTA_SSL_EXT.1	TSF-initiated Session Locking
	FTA_SSL.3	TSF-initiated Termination
	FTA_SSL.4	User-initiated Termination
	FTA_TAB.1	Default TOE Access Banners
FTP: Trusted path/channels	FTP_ITC.1	Inter-TSF trusted channel (Refinement)

Class Name	Component Identification	Component Name
	FTP_TRP.1/Admin	Trusted Path (Refinement)

5.3 SFRs from NDcPP

5.3.1 Security audit (FAU)

5.3.1.1 FAU_GEN.1 Audit data generation

FAU_GEN.1.1 The TSF shall be able to generate an audit record of the following auditable events:

- a) Start-up and shut-down of the audit functions;
- b) All auditable events for the not specified level of audit; and
- c) *All administrator actions comprising:*
 - *Administrative login and logout (name of user account shall be logged if individual user accounts are required for administrators).*
 - *Changes to TSF data related to configuration changes (in addition to the information that a change occurred it shall be logged what has been changed).*
 - *Generating/import of, changing, or deleting of cryptographic keys (in addition to the action itself a unique key name or key reference shall be logged).*
 - *Resetting passwords (name of related user account shall be logged).*
 - *[[no other actions]];*
- d) *Specifically defined auditable events listed in Table 16.*

FAU_GEN.1.2 The TSF shall record within each audit record at least the following information:

- a) Date and time of the event, type of event, subject identity, and the outcome (success or failure) of the event; and
- b) For each audit event type, based on the auditable event definitions of the functional components included in the cPP/ST, *information specified in column three of Table 16.*

Table 16 Auditable Events

SFR	Auditable Event	Additional Audit Record Contents
FAU_GEN.1	None.	None.
FAU_GEN.2	None.	None.
FAU_STG_EXT.1	None.	None.
FCS_CKM.1	None.	None.
FCS_CKM.2	None.	None.
FCS_CKM.4	None.	None.
FCS COP.1/DataEncryption	None.	None.

SFR	Auditable Event	Additional Audit Record Contents
FCS_COP.1/SigGen	None.	None.
FCS_COP.1/Hash	None.	None.
FCS_COP.1/KeyedHash	None.	None.
FCS_IPSEC_EXT.1	Failure to establish an IPsec SA.	Reason for failure.
FCS_RBG_EXT.1	None.	None.
FCS_SSHS_EXT.1	Failure to establish an SSH session	Reason for failure.
FIA_AFL.1	Unsuccessful login attempts limit is met or exceeded.	Origin of the attempt (e.g., IP address).
FIA_PMG_EXT.1	None.	None.
FIA_UIA_EXT.1	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU_EXT.2	All use of the identification and authentication mechanism.	Origin of the attempt (e.g., IP address).
FIA_UAU.7	None.	None.
FIA_X509_EXT.1/Rev	Unsuccessful attempt to validate a certificate	Reason for failure
FIA_X509_EXT.2	None.	None.
FIA_X509_EXT.3	None.	None.
FMT_MOF.1/Manual Update	Any attempt to initiate a manual update	None.
FMT_MTD.1/CoreData	All management activities of TSF data	None.
FMT_SMF.1	None.	None.
FMT_SMR.2	None.	None.
FPT_SKP_EXT.1	None.	None.
FPT_APW_EXT.1	None.	None.
FPT_STM_EXT.1	Discontinuous changes to time - either Administrator actuated or changed via an automated process. (Note that no	For discontinuous changes to time: The old and new values for the time.

SFR	Auditable Event	Additional Audit Record Contents
	continuous changes to time need to be logged. See also application note on FPT_STM_EXT.1)	Origin of the attempt to change time for success and failure (e.g., IP address).
FPT_TUD_EXT.1	Initiation of update. result of the update attempt (success or failure)	No additional information.
FPT_TST_EXT.1	None.	None.
FTA_SSL_EXT.1	The termination of a local session by the session locking mechanism.	None.
FTA_SSL.3	The termination of a remote session by the session locking mechanism.	None.
FTA_SSL.4	The termination of an interactive session.	None.
FTA_TAB.1	None.	None.
FTP_ITC.1	Initiation of the trusted channel. Termination of the trusted channel. Failure of the trusted channel functions.	Identification of the initiator and target of failed trusted channels establishment attempt
FTP_TRP.1/Admin	Initiation of the trusted path. Termination of the trusted path. Failure of the trusted path functions.	None.

5.3.1.2 FAU_GEN.2 User Identity Association

FAU_GEN.2.1 For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

5.3.1.3 FAU_STG_EXT.1 Protected Audit Event Storage

FAU_STG_EXT.1.1 The TSF shall be able to transmit the generated audit data to an external IT entity using a trusted channel according to FTP_ITC.1.

FAU_STG_EXT.1.2 The TSF shall be able to store generated audit data on the TOE itself.

FAU_STG_EXT.1.3 The TSF shall [overwrite previous audit records according to the following rule: *[the newest audit record will overwrite the oldest audit record.]*] when the local storage space for audit data is full.

5.3.2 Cryptographic Support (FCS)

5.3.2.1 FCS_CKM.1 Cryptographic Key Generation (Refined)

FCS_CKM.1.1 The TSF shall generate **asymmetric** cryptographic keys in accordance with a specified cryptographic key generation algorithm: [

- RSA schemes using cryptographic key sizes of 2048-bit or greater that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.3
- ECC schemes using “NIST curves” [P-256, P-384] that meet the following: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Appendix B.4;
- FFC Schemes using Diffie-Hellman group 14 that meet the following: RFC 3526, Section 3

] and specified cryptographic key sizes [assignment: cryptographic key sizes] that meet the following: [assignment: list of standards].

5.3.2.2 FCS_CKM.2 Cryptographic Key Establishment (Refined)

FCS_CKM.2.1 The TSF shall **perform** cryptographic **key establishment** in accordance with a specified cryptographic key **establishment** method: [

- Elliptic curve-based key establishment schemes that meets the following: NIST Special Publication 800-56A Revision 2, “Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography”;
- Key establishment scheme using Diffie-Hellman group 14 that meets the following: RFC 3526, Section 3;

] that meets the following: [assignment: list of standards].

5.3.2.3 FCS_CKM.4 Cryptographic Key Destruction

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method

- *For plaintext keys in volatile storage, the destruction shall be executed by a [single overwrite consisting of [zeroes]];*
- *For plaintext keys in non-volatile storage, the destruction shall be executed by the invocation of an interface provided by a part of the TSF that [*
 - logically addresses the storage location of the key and performs a [single-pass] overwrite consisting of [zeroes];

that meets the following: No Standard.

5.3.2.4 FCS_COP.1/DataEncryption Cryptographic Operation (AES Data Encryption/Decryption)

FCS_COP.1.1/DataEncryption The TSF shall perform *encryption/decryption* in accordance with a specified cryptographic algorithm *AES used in [CBC] mode* and cryptographic key sizes [128 bits, 192 bits, 256 bits] that meet the following: *AES as specified in ISO 18033-3, [CBC as specified in ISO 10116]*.

5.3.2.5 FCS_COP.1/SigGen Cryptographic Operation (Signature Generation and Verification)

FCS_COP.1.1/SigGen The TSF shall perform *cryptographic signature services (generation and verification)* in accordance with a specified cryptographic algorithm[

- RSA Digital Signature Algorithm and cryptographic key sizes (modulus) [2048 bits or greater],
- Elliptic Curve Digital Signature Algorithm and cryptographic key sizes [256 bits or greater]

] and cryptographic key sizes [~~assignment: cryptographic key sizes~~]

that meet the following: [

- For RSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 5.5, using PKCS #1 v2.1 Signature Schemes RSASSA-PSS and/or RSASSAPKCS1v1_5; ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3,
- For ECDSA schemes: FIPS PUB 186-4, “Digital Signature Standard (DSS)”, Section 6 and Appendix D, Implementing “NIST curves” [P-256, P-384]; ISO/IEC 14888-3, Section 6.4

].

5.3.2.6 FCS_COP.1/Hash Cryptographic Operation (Hash Algorithm)

FCS_COP.1.1/Hash The TSF shall perform *cryptographic hashing services* in accordance with a specified cryptographic algorithm [SHA-1, SHA-256, SHA-512] and ~~cryptographic key sizes [assignment: cryptographic key sizes]~~ and **message digest sizes [160, 256, 512]** bits that meet the following: *ISO/IEC 10118-3:2004*.

5.3.2.7 FCS_COP.1/KeyedHash Cryptographic Operation (Keyed Hash Algorithm)

FCS_COP.1.1/KeyedHash The TSF shall perform *keyed-hash message authentication* in accordance with a specified cryptographic algorithm [HMAC-SHA-1] and cryptographic key sizes [160-bit] and **message digest sizes [160] bits** that meet the following: *ISO/IEC 9797-*

2:2011, Section 7 “MAC Algorithm 2”.

5.3.2.8 FCS_IPSEC_EXT.1 IPsec Protocol

FCS_IPSEC_EXT.1.1 The TSF shall implement the IPsec architecture as specified in RFC 4301.

FCS_IPSEC_EXT.1.2 The TSF shall have a nominal, final entry in the SPD that matches anything that is otherwise unmatched, and discards it.

FCS_IPSEC_EXT.1.3 The TSF shall implement [transport mode, tunnel mode].

FCS_IPSEC_EXT.1.4 The TSF shall implement the IPsec protocol ESP as defined by RFC 4303 using the cryptographic algorithms [AES-CBC-128, AES-CBC-192, AES-CBC-256 (specified by RFC 3602)] together with a Secure Hash Algorithm (SHA)-based HMAC [HMAC-SHA1] and [no other algorithm].

FCS_IPSEC_EXT.1.5 The TSF shall implement the protocol: [

- IKEv1, using Main Mode for Phase 1 exchanges, as defined in RFCs 2407, 2408, 2409, RFC 4109, [no other RFCs for extended sequence numbers], and [no other RFCs for hash functions].
- IKEv2 as defined in RFC 5996 and [with mandatory support or NAT traversal as specified in RFC 5996, section 2.23)], and [RFC 4868 for hash functions]

].

FCS_IPSEC_EXT.1.6 The TSF shall ensure the encrypted payload in the [IKEv1, IKEv2] protocol uses the cryptographic algorithms [AES-CBC-128, AES-CBC-192, AES-CBC-256 (as specified in RFC 3602)].

FCS_IPSEC_EXT.1.7 The TSF shall ensure that [

- IKEv1 Phase 1 SA lifetimes can be configured by an Security Administrator based on

[

- length of time, where the time values can configured within [1-24] hours;

].

- IKEv2 SA lifetimes can be configured by an Security Administrator based on

[

- length of time, where the time values can configured within [1-24] hours;

].

]

FCS_IPSEC_EXT.1.8 The TSF shall ensure that [

- IKEv1 Phase 2 SA lifetimes can be configured by an Security Administrator based on

[

- number of bytes
- length of time, where the time values can configured within [1-8] hours;

].

- IKEv2 Child SA lifetimes can be configured by an Security Administrator based on
 - number of bytes
 - length of time, where the time values can be configured within [1-8] hours;

].

]

FCS_IPSEC_EXT.1.9 The TSF shall generate the secret value x used in the IKE Diffie-Hellman key exchange (“ x ” in $g^x \text{ mod } p$) using the random bit generator specified in FCS_RBG_EXT.1, and having a length of at least [112 (for DH Group 14), 128 (for DH Group 19), 128 (for DH Group 24), and 192 (for DH Group 20)] bits.

FCS_IPSEC_EXT.1.10 The TSF shall generate nonces used in [IKEv1, IKEv2] exchanges of length [

- according to the security strength associated with the negotiated Diffie-Hellman group;
- at least 128 bits in size and at least half the output size of the negotiated pseudorandom function (PRF) hash

].

FCS_IPSEC_EXT.1.11 The TSF shall ensure that all IKE protocols implement DH Group(s) [14 (2048-bit MODP), 19 (256-bit Random ECP), 20(384-bit Random ECP), and 24 (2048-bit MODP with 256-bit POS)].

FCS_IPSEC_EXT.1.12 The TSF shall be able to ensure by default that the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 1, IKEv2 IKE_SA] connection is greater than or equal to the strength of the symmetric algorithm (in terms of the number of bits in the key) negotiated to protect the [IKEv1 Phase 2, IKEv2 CHILD_SA] connection.

FCS_IPSEC_EXT.1.13 The TSF shall ensure that all IKE protocols perform peer authentication using [RSA, ECDSA] that use X.509v3 certificates that conform to RFC 4945 and [Pre-shared Keys].

FCS_IPSEC_EXT.1.14 The TSF shall only establish a trusted channel if the presented identifier in the received certificate matches the configured reference identifier, where the presented and reference identifiers are of the following fields and types: [CN: IP address, CN: Fully Qualified Domain Name (FQDN), CN: user FQDN, Distinguished Name (DN)] and [no other reference identifier type]].

5.3.2.9 FCS_RBG_EXT.1 Extended: Cryptographic Operation (Random Bit Generation)

FCS_RBG_EXT.1.1 The TSF shall perform all deterministic random bit generation services in accordance with ISO/IEC 18031:2011 using [CTR_DRBG (AES)].

FCS_RBG_EXT.1.2 The deterministic RBG shall be seeded by at least one entropy source that accumulates entropy from [[1] hardware-based noise source] with a minimum of [256 bits] of entropy at least equal to the greatest security strength, according to ISO/IEC 18031:2011 Table C.1 “Security Strength Table for Hash Functions”, of the keys and hashes that it will generate.

5.3.2.10 FCS_SSHS_EXT.1 SSH Server Protocol

FCS_SSHS_EXT.1.1 The TSF shall implement the SSH protocol that complies with RFCs [4251, 4252, 4253, 4254].

FCS_SSHS_EXT.1.2 The TSF shall ensure that the SSH protocol implementation supports the following authentication methods as described in RFC 4252: public key-based, [password based].

FCS_SSHS_EXT.1.3 The TSF shall ensure that, as described in RFC 4253, packets greater than [65,535] bytes in an SSH transport connection are dropped.

FCS_SSHS_EXT.1.4 The TSF shall ensure that the SSH transport implementation uses the following encryption algorithms and rejects all other encryption algorithms: [aes128-cbc, aes256-cbc].

FCS_SSHS_EXT.1.5 The TSF shall ensure that the SSH public-key based authentication implementation uses [ssh-rsa] as its public key algorithm(s) and rejects all other public key algorithms.

FCS_SSHS_EXT.1.6 The TSF shall ensure that the SSH transport implementation uses [hmac-sha1, hmac-sha1-96] as its data integrity MAC algorithm(s) and rejects all other MAC algorithm(s).

FCS_SSHS_EXT.1.7 The TSF shall ensure that [diffie-hellman-group14-sha1] are the only allowed key exchange methods used for the SSH protocol.

FCS_SSHS_EXT.1.8 The TSF shall ensure that within SSH connections the same session keys are used for a threshold of no longer than one hour, and no more than one gigabyte of transmitted data. After either of the thresholds are reached a rekey needs to be performed.

5.3.3 Identification and authentication (FIA)

5.3.3.1 FIA_AFL.1 Authentication Failure Handling

FIA_AFL.1.1 The TSF shall detect when an Administrator configurable positive integer within [1 to 25] unsuccessful authentication attempts occur related to *Administrators attempting to authenticate remotely using a password*.

FIA_AFL.1.2 When the defined number of unsuccessful authentication attempts has been met, the TSF shall [prevent the offending Administrator from successfully establishing remote session using any authentication method that involves a password until [an authorized administrator unlocks the locked user account] is taken by an Administrator].

5.3.3.2 FIA_PMG_EXT.1 Password Management

FIA_PMG_EXT.1.1 The TSF shall provide the following password management capabilities for administrative passwords:

- a) Passwords shall be able to be composed of any combination of upper and lower case letters, numbers, and the following special characters: [“!” , “@” , “#” , “\$” , “%” , “^” , “&” , “*” , “(” , “)”];
- b) Minimum password length shall be configurable to [15] and [15].

5.3.3.3 FIA_UIA_EXT.1 User Identification and Authentication

FIA_UIA_EXT.1.1 The TSF shall allow the following actions prior to requiring the non-TOE entity to initiate the identification and authentication process:

- Display the warning banner in accordance with FTA_TAB.1;
- [no other actions]

FIA_UIA_EXT.1.2 The TSF shall require each administrative user to be successfully identified and authenticated before allowing any other TSF-mediated action on behalf of that administrative user.

5.3.3.4 FIA_UAU_EXT.2 Extended: Password-based Authentication Mechanism

FIA_UAU_EXT.2.1 The TSF shall provide a local password-based authentication mechanism, and [remote password-based authentication via RADIUS] to perform local administrative user authentication.

5.3.3.5 FIA_UAU.7 Protected Authentication Feedback

FIA_UAU.7.1 The TSF shall provide only *obscured feedback* to the administrative user while the authentication is in progress **at the local console**.

5.3.3.6 FIA_X509_EXT.1/Rev X.509 Certificate Validation

FIA_X509_EXT.1.1/Rev The TSF shall validate certificates in accordance with the following rules:

- RFC 5280 certificate validation and certificate path validation **supporting a minimum path length of three certificates**.
- The certificate path must terminate with a trusted CA certificate.
- The TSF shall validate a certification path by ensuring that all CA certificates in the certification path contain the basicConstraints extension with the CA flag set to TRUE.

- The TSF shall validate the revocation status of the certificate using [a Certificate Revocation List (CRL) as specified in RFC5759 Section 5].
- The TSF shall validate the extendedKeyUsage field according to the following rules:
 - *Certificates used for trusted updates and executable code integrity verification shall have the Code Signing purpose (id-kp 3 with OID 1.3.6.1.5.5.7.3.3) in the extendedKeyUsage field.*
 - *Server certificates presented for TLS shall have the Server Authentication purpose (id-kp 1 with OID 1.3.6.1.5.5.7.3.1) in the extendedKeyUsage field.*
 - *Client certificates presented for TLS shall have the Client Authentication purpose (id-kp 2 with OID 1.3.6.1.5.5.7.3.2) in the extendedKeyUsage field.*
 - *OCSP certificates presented for OCSP responses shall have the OCSP Signing purpose (id-kp 9 with OID 1.3.6.1.5.5.7.3.9) in the extendedKeyUsage field.*

FIA_X509_EXT.1.2/Rev The TSF shall only treat a certificate as a CA certificate if the basicConstraints extension is present and the CA flag is set to TRUE.

5.3.3.7 FIA_X509_EXT.2 X.509 Certificate Authentication

FIA_X509_EXT.2.1 The TSF shall use X.509v3 certificates as defined by RFC 5280 to support authentication for [IPsec], and [no additional uses].

FIA_X509_EXT.2.2 When the TSF cannot establish a connection to determine the validity of a certificate, the TSF shall [allow the administrator to choose whether to accept the certificate in these cases].

5.3.3.8 FIA_X509_EXT.3 X.509 Certificate Requests

FIA_X509_EXT.3.1 The TSF shall generate a Certificate Request as specified by RFC 2986 and be able to provide the following information in the request: public key and [Common Name, Organization, Organizational Unit, Country].

FIA_X509_EXT.3.2 The TSF shall validate the chain of certificates from the Root CA upon receiving the CA Certificate Response.

5.3.4 Security management (FMT)

5.3.4.1 FMT_MOF.1/ManualUpdate Management of security functions behavior

FMT_MOF.1/ManualUpdate The TSF shall restrict the ability to enable the functions *to perform manual update to Security Administrators*.

5.3.4.1 FMT_MTD.1/CoreData Management of TSF Data

FMT_MTD.1/CoreData The TSF shall restrict the ability to manage the *TSF data to Security Administrators*.

5.3.4.2 FMT_SMF.1 Specification of Management Functions

FMT_SMF.1.1 The TSF shall be capable of performing the following management functions:

- *Ability to administer the TOE locally and remotely;*
- *Ability to configure the access banner;*
- *Ability to configure the session inactivity time before session termination or locking;*
- *Ability to update the TOE, and to verify the updates using [digital signature, hash comparison] capability prior to installing those updates;*
- *Ability to configure the authentication failure parameters for FIA_AFL.1;*
- [
 - *Ability to configure audit behavior;*
 - *Ability to configure the cryptographic functionality;*
 - *Ability to configure thresholds for SSH rekeying;*
 - *Ability to configure the lifetime for IPsec SAs;*
 - *Ability to re-enable an Administrator account;*
 - *Ability to set the time which is used for time-stamps;*
 - *Ability to configure the reference identifier for the peer;*

5.3.4.3 FMT_SMR.2 Restrictions on Security Roles

FMT_SMR.2.1 The TSF shall maintain the roles:

- *Security Administrator.*

FMT_SMR.2.2 The TSF shall be able to associate users with roles.

FMT_SMR.2.3 The TSF shall ensure that the conditions

- *The Security Administrator role shall be able to administer the TOE locally;*
- *The Security Administrator role shall be able to administer the TOE remotely*

5.3.5 Protection of the TSF (FPT)

5.3.5.1 FPT_SKP_EXT.1: Protection of TSF Data (for reading of all pre-shared, symmetric and private keys)

FPT_SKP_EXT.1.1 The TSF shall prevent reading of all pre-shared keys, symmetric keys, and private keys.

5.3.5.2 FPT_APW_EXT.1: Protection of Administrator Passwords

FPT_APW_EXT.1.1 The TSF shall store passwords in non-plaintext form.

FPT_APW_EXT.1.2 The TSF shall prevent the reading of plaintext passwords.

5.3.5.3 FPT_STM_EXT.1 Reliable Time Stamps

FPT_STM_EXT.1.1 The TSF shall be able to provide reliable time stamps for its own use.

FPT_STM_EXT.1.2 The TSF shall [allow the Security Administrator to set the time].

5.3.5.4 FPT_TST_EXT.1: TSF Testing

FPT_TST_EXT.1.1 The TSF shall run a suite of the following self-tests [during initial start-up (on power on), periodically during normal operation] to demonstrate the correct operation of the TSF: [

- *AES Known Answer Test*
- *RSA Signature Known Answer Test (both signature/verification)*
- *RNG/DRBG Known Answer Test*
- *HMAC Known Answer Test*
- *SHA-1/256/512 Known Answer Test*
- *ECDSA self-test*
- *Software Integrity Test*

].

5.3.5.5 FPT_TUD_EXT.1 Extended: Trusted Update

FPT_TUD_EXT.1.1 The TSF shall provide *Security Administrators* the ability to query the currently executing version of the TOE firmware/software and [no other TOE firmware/software version].

FPT_TUD_EXT.1.2 The TSF shall provide *Security Administrators* the ability to manually initiate updates to TOE firmware/software and [no other update mechanism].

FPT_TUD_EXT.1.3 The TSF shall provide a means to authenticate firmware/software updates to the TOE using a [digital signature mechanism, published hash] prior to installing those updates.

5.3.6 TOE Access (FTA)

5.3.6.1 FTA_SSL_EXT.1 TSF-initiated Session Locking

FTA_SSL_EXT.1.1 The TSF shall, for local interactive sessions, [

- terminate the session]

after a Security Administrator-specified time period of inactivity.

5.3.6.2 FTA_SSL.3 TSF-initiated Termination

FTA_SSL.3.1: The TSF shall terminate a **remote** interactive session after a *Security Administrator-configurable time interval of session inactivity*.

5.3.6.3 FTA_SSL.4 User-initiated Termination

FTA_SSL.4.1 The TSF shall allow **Administrator**-initiated termination of the **Administrator's** own interactive session.

5.3.6.4 FTA_TAB.1 Default TOE Access Banners

FTA_TAB.1.1: Before establishing an **administrative user** session the TSF shall display a **Security Administrator-specified advisory notice and consent** warning message regarding use of the TOE.

5.3.7 Trusted Path/Channels (FTP)

5.3.7.1 FTP_ITC.1 Inter-TSF trusted channel

FTP_ITC.1.1 The TSF shall **be capable of using [IPsec] to** provide a trusted communication channel between itself and **authorized IT entities supporting the following capabilities: audit server, [authentication server, [no other capabilities]]** that is logically distinct from other communication channels and provides assured identification of its end points and protection of the channel data from **disclosure and detection of modification of the channel data**.

FTP_ITC.1.2 The TSF shall permit the TSF or the authorized IT entities to initiate communication via the trusted channel.

FTP_ITC.1.3 The TSF shall initiate communication via the trusted channel for [communications with the following:

- external audit servers using IPsec,
- remote AAA servers using IPsec

].

5.3.7.2 FTP_TRP.1/Admin Trusted Path

FTP_TRP.1.1/Admin: The TSF shall be capable of using **[SSH]** to provide a communication path between itself and **authorized remote administrators** that is logically distinct from other communication paths and provides assured identification of its end points and protection of the communicated data from **disclosure and provides detection of modification of the channel data**.

FTP_TRP.1.2/Admin The TSF shall permit remote Administrators to initiate communication via the trusted path.

FTP_TRP.1.3/Admin The TSF shall require the use of the trusted path for initial Administrator authentication and all remote administration actions.

5.4 TOE SFR Dependencies Rationale for SFRs Found in PP

The NDcPP v2.0e contain all the requirements claimed in this Security Target. As such the dependencies are not applicable since the PP have been approved.

5.5 Security Assurance Requirements

5.5.1 SAR Requirements

The TOE assurance requirements for this ST are taken directly from the NDcPP which are derived from Common Criteria Version 3.1, Revision 4. The assurance requirements are summarized in the table below.

Table 17 Assurance Measures

Assurance Class	Components	Components Description
Security Target (ASE)	ASE_CCL.1	Conformance claims
	ASE_ECD.1	Extended components definition
	ASE_INT.1	ST introduction
	ASE_OBJ.1	Security objectives for the operational environment
	ASE_REQ.1	Stated security requirements
	ASE_SPD.1	Security Problem Definition
	ASE_TSS.1	TOE summary specification
Development (ADV)	ADV_FSP.1	Basic Functional Specification
Guidance documents (AGD)	AGD_OPE.1	Operational user guidance
	AGD_PRE.1	Preparative User guidance
Life cycle support (ALC)	ALC_CMC.1	Labeling of the TOE
	ALC_CMS.1	TOE CM coverage
Tests (ATE)	ATE_IND.1	Independent testing - sample
Vulnerability assessment (AVA)	AVA_VAN.1	Vulnerability survey

5.5.2 Security Assurance Requirements Rationale

The Security Assurance Requirements (SARs) in this Security Target represent the SARs identified in the NDcPPv2.0e. As such, the NDcPP SAR rationale is deemed acceptable since the PPs have been validated.

5.6 Assurance Measures

The TOE satisfies the identified assurance requirements. This section identifies the Assurance Measures applied by Cisco to satisfy the assurance requirements. The table below lists the details.

Table 18 Assurance Measures

Component	How requirement will be met
ADV_FSP.1	The functional specification describes the external interfaces of the TOE; such as the means for a user to invoke a service and the corresponding response of those services. The description includes the interface(s) that enforces a security functional requirement, the interface(s) that supports the enforcement of a security functional requirement, and the interface(s) that does not enforce any security functional requirements. The interfaces are described in terms of their purpose (general goal of the interface), method of use (how the interface is to be used), parameters (explicit inputs to and outputs from an interface that control the behavior of that interface), parameter descriptions (tells what the parameter is in some meaningful way), and error messages (identifies the condition that generated it, what the message is, and the meaning of any error codes). The development evidence also contains a tracing of the interfaces to the SFRs described in this ST.
AGD_OPE.1	The Administrative Guide provides the descriptions of the processes and procedures of how the administrative users of the TOE can securely administer the TOE using the interfaces that provide the features and functions detailed in the guidance.
AGD_PRE.1	The Installation Guide describes the installation, generation, and startup procedures so that the users of the TOE can put the components of the TOE in the evaluated configuration.
ALC_CMC.1	The Configuration Management (CM) document(s) describes how the consumer (end-user) of the TOE can identify the evaluated TOE (Target of Evaluation). The CM document(s), identifies the configuration items, how those configuration items are uniquely identified, and the adequacy of the procedures that are used to control and track changes that are made to the TOE. This includes details on what changes are tracked, how potential changes are incorporated, and the degree to which automation is used to reduce the scope for error. The TOE will also be provided along with the appropriate administrative guidance.
ALC_CMS.1	
ATE_IND.1	Cisco will provide the TOE for testing.
AVA_VAN.1	Cisco will provide the TOE for testing.

6 TOE SUMMARY SPECIFICATION

6.1 TOE Security Functional Requirement Measures

This chapter identifies and describes how the Security Functional Requirements identified above are met by the TOE.

Table 19 How TOE SFRs Measures

TOE SFRs	How the SFR is Met
FAU_GEN.1	<p>The TOE generates an audit record whenever an audited event occurs. The types of events that cause audit records to be generated include: startup and shutdown of the audit mechanism cryptography related events, identification and authentication related events, and administrative events (the specific events and the contents of each audit record are listed in the table within the FAU_GEN.1 SFR, “Auditable Events Table”). Each of the events is specified in syslog records in enough detail to identify the user for which the event is associated, when the event occurred, where the event occurred, the outcome of the event, and the type of event that occurred such as generating keys, including the type of key. Additionally, the startup and shutdown of the audit functionality is audited.</p> <p>The audit trail consists of the individual audit records; one audit record for each event that occurred. The audit record can contain up to 80 characters and a percent sign (%), which follows the time-stamp information. As noted above, the information includes at least all of the required information. Example audit events are included below:</p> <pre>Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test activated by user: lab) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (Software checksum ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (DES encryption/decryption ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (3DES encryption/decryption ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (SHA hashing ... passed) Nov 19 13:55:59: %CRYPTO-6-SELF_TEST_RESULT: Self test info: (AES encryption/decryption ... passed)</pre> <p>In the above log events a date and timestamp is displayed as well as an event description “CRYPTO-6-SELF_TEST_RESULT: Self test info: (Self test)”. The subject identity where a command is directly run by a user is displayed “user: lab.” The outcome of the command is displayed: “passed”</p> <p>The logging buffer size can be configured from a range of 4096 (default) to 4,294,967,295 bytes. It is noted to not make the buffer size too large because the TOE could run out of memory for other tasks. Use the show memory privileged EXEC command to view the free processor memory on the TOE. However, this value is the maximum available, and the buffer size should not be set to this amount.</p> <p>The administrator can also configure a ‘configuration logger’ to keep track of configuration changes made with the command-line interface (CLI). The</p>

TOE SFRs	How the SFR is Met
	<p>administrator can configure the size of the configuration log from 1 to 1000 entries (the default is 100).</p> <p>The log buffer is circular, so newer messages overwrite older messages after the buffer is full. Administrators are instructed to monitor the log buffer using the show logging privileged EXEC command to view the audit records. The first message displayed is the oldest message in the buffer. There are other associated commands to clear the buffer, to set the logging level, etc. The logs can be saved to flash memory so records are not lost in case of failures or restarts. Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.</p> <p>The administrator can set the level of the audit records to be displayed on the console or sent to the syslog server. For instance all emergency, alerts, critical, errors, and warning messages can be sent to the console alerting the administrator that some action needs to be taken as these types of messages mean that the functionality of the TOE is affected. All notifications and information type message can be sent to the syslog server. The audit records are transmitted using IPsec tunnel to the syslog server. If the communications to the syslog server is lost, the TOE generates an audit record and all permit traffic is denied until the communications is re-established.</p> <p>Once the box is up and operational and the crypto self test command is entered, then the result messages would be displayed on the console and will also be logged. If the TOE encounters a failure to invoke any one of the cryptographic functions, a log record is generated.</p> <p>When the incoming traffic to the TOE exceeds what the interface can handle, the packets are dropped at the input queue itself and there are no error messages generated.</p>
FAU_GEN.2	<p>The TOE shall ensure that each auditable event is associated with the user that triggered the event and as a result, they are traceable to a specific user. For example, a human user, user identity or related session ID would be included in the audit record. For an IT entity or device, the IP address, MAC address, host name, or other configured identification is presented. A sample audit record is below:</p> <pre>Jun 18 11:17:20.769: AAA/BIND(0000004B): Bind i/f Jun 18 11:17:20.769: AAA/AUTHEN/LOGIN (0000004B): Pick method list 'default' Jun 18 2012 11:17:26 UTC: %SEC_LOGIN-5-LOGIN_SUCCESS: Login Success [user: admin] [Source: 100.1.1.5] [localport: 22] at 11:17:26 UTC Mon Jun 18 2012</pre>
FAU_STG_EXT.1	<p>The TOE is configured to export syslog records to a specified, external syslog server in real-time. The TOE protects communications with an external syslog server via IPsec. If the IPsec connection fails, the TOE will store audit records on the TOE when it discovers it can no longer communicate with its configured syslog server. When the connection is restored, the TOE will transmit the buffer contents when connectivity to the syslog server.</p> <p>For audit records stored internally to the TOE the audit records are stored in a circular log file where the TOE overwrites the oldest audit records when the audit trail becomes full. The size of the logging files on the TOE is</p>

TOE SFRs	How the SFR is Met
	<p>configurable by the administrator with the minimum value being 4096 (default) to 2147483647 bytes of available disk space Refer to the Common Criteria Operational User Guidance and Preparative Procedures for command description and usage information.</p> <p>Only Authorized Administrators are able to clear the local logs, and local audit records are stored in a directory that does not allow administrators to modify the contents.</p>
FCS_CKM.1	<p>The TOE implements DH group 14 key establishment schemes that meets RFC 3526, Section 3 and Elliptic curve key establishment (conformant to NIST SP 800-56A). The TOE acts as both a sender and receiver for Diffie-Helman based key establishment schemes.</p>
FCS_CKM.2	<p>The TOE complies with section 5.6 and all subsections regarding asymmetric key pair generation and key establishment in the NIST SP 800-56A and with section 6 and all subsections regarding RSA key pair generation and key establishment in the NIST SP 800-56B.</p> <p>Asymmetric cryptographic keys used for IKE peer authentication are generated according to FIPS PUB 186-4, Appendix B.3 for RSA schemes and Appendix B.4 for ECDSA schemes.</p> <p>The TOE can create an RSA public-private key pair, with a minimum RSA key size of 2048-bit and ECDSA key pairs using NIST curves P-256 and P-384. Both RSA and ECC schemes can be used to generate a Certificate Signing Request (CSR).</p> <p>Through use of Simple Certificate Enrollment Protocol (SCEP), the TOE can: send the CSR to a Certificate Authority (CA) for the CA to generate a certificate; and receive its X.509v3 certificate from the CA. Integrity of the CSR and certificate during transit are assured through use of digitally signatures (encrypting the hash of the TOE's public key contained in the CSR and certificate). The TOE can store and distribute the certificate to external entities including Registration Authorities (RA). The IOS-XE Software supports embedded PKI client functions that provide secure mechanisms for distributing, managing, and revoking certificates. In addition, the IOS-XE Software includes an embedded certificate server, allowing the router to act as a certification authority on the network.</p> <p>The TOE can also use the X.509v3 certificate for securing IPsec. The TOE provides cryptographic signature services using ECDSA that meets FIPS 186-4, "Digital Signature Standard" with NIST curves P-256, P-384 and RSA that meets FIPS PUB 186-4, "Digital Signature Standard".</p>
FCS_CKM.4	<p>The TOE meets all requirements specified in FIPS 140-2 for destruction of keys and Critical Security Parameters (CSPs). Refer to Table 20 for more information on the key zeroization.</p>
FCS_COP.1/DataEncryption	<p>The TOE provides symmetric encryption and decryption capabilities using AES in CBC mode (128, 192, 256 bits) as described in ISO 18033-3 and ISO 10116. AES is implemented in the following protocols: IPsec and SSH. The relevant FIPS certificate numbers are listed in Table 6.</p>
FCS_COP.1/SigGen	
FCS_COP.1/Hash	
FCS_COP.1/KeyedHash	

TOE SFRs	How the SFR is Met
	<p>The TOE provides cryptographic signature services using RSA Digital Signature Algorithm with key size of 2048 and greater as specified in ISO/IEC 9796-2, Digital signature scheme 2 or Digital Signature scheme 3. In addition, the TOE will provide cryptographic signature services using ECDSA with key size of 256 or greater as specified in FIPS PUB 186-4, “Digital Signature Standard”.</p> <p>The TOE provides cryptographic hashing services using SHA-1, SHA-256 and SHA-512 as specified in ISO/IEC 10118-3:2004. The TOE provides keyed-hashing message authentication services using HMAC-SHA-1 operates on 512-bit blocks with key sizes and message digest sizes of 160-bits as specified in ISO/IEC 9797-2:2011, Section 7 “MAC Algorithm 2”.</p> <p>Passwords are stored as a SHA-256 hash with the use of the “service password-encryption” command in config mode.</p> <p>For IKE (ISAKMP) hashing, administrators can select HMAC-SHA-1 (with message digest sizes of 160) to be used with remote IPsec endpoints.</p> <p>SHA-512 hashing is used for verification of software image integrity. The relevant FIPS certificate numbers are listed in Table 6 FIPS References.</p> <p>The TOE provides Secure Hash Standard (SHS) hashing in support of SSH for secure communications. Management of the cryptographic algorithms is provided through the CLI with auditing of those commands.</p> <p>The TOE uses HMAC-SHA1 message authentication as part of the RADIUS Key Wrap functionality. For IPsec SA authentication integrity options administrators can select HMAC-SHA-1 (with message digest sizes of 160) to be part of the IPsec SA transform-set to be used with remote IPsec endpoints.</p> <p>The relevant FIPS certificate numbers are listed in Table 6.</p>
FCS_IPSEC_EXT.1	<p>The TOE implements IPsec to provide authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network.</p> <p>The IPsec implementation provides both VPN peer-to-peer and VPN client to TOE capabilities. The VPN peer-to-peer tunnel allows for example the TOE and another switch to establish an IPsec tunnel to secure the passing of route tables (user data). Another configuration in the peer-to-peer configuration is to have the TOE be set up with an IPsec tunnel with a VPN peer to secure the session between the TOE and syslog server. The VPN client to TOE configuration would be where a remote VPN client connects into the TOE in order to gain access to an authorized private network. Authenticating with the TOE would give the VPN client a secure IPsec tunnel to connect over the internet into their private network.</p> <p>In addition to tunnel mode, which is the default IPsec mode, the TOE also supports transport mode, allowing for only the payload of the packet to be encrypted. If tunnel mode is explicitly specified, the switch will request tunnel mode and will accept only tunnel mode.</p>

TOE SFRs	How the SFR is Met
	<p>The TOE implements IPsec to provide both certificates and pre-shared key-based authentication and encryption services to prevent unauthorized viewing or modification of data as it travels over the external network.</p> <p>Preshared keys can be configured using the 'crypto isakmp key' key command and may be proposed by each of the peers negotiating the IKE establishment.</p> <p>IPsec Internet Key Exchange, also called ISAKMP, is the negotiation protocol that lets two peers agree on how to build an IPsec Security Association (SA). The IKE protocols implement Peer Authentication using the RSA and ECDSA algorithm with X.509v3 certificates or preshared keys.</p> <p>When certificates are used for authentication, the distinguished name (DN) is verified to ensure the certificate is valid and is from a valid entity. The DN naming attributes in the certificate is compared with the expected DN naming attributes and deemed valid if the attribute types are the same and the values are the same and as expected. The fully qualified domain name (FQDN) can also be used as verification where the attributes in the certificate are compared with the expected CN: FQDN, CN: user FQDN and CN: IP Address.</p> <p>IKE separates negotiation into two phases: phase 1 and phase 2. Phase 1 creates the first tunnel, which protects later ISAKMP negotiation messages. The key negotiated in phase 1 enables IKE peers to communicate securely in phase 2. During Phase 2 IKE establishes the IPsec SA. IKE maintains a trusted channel, referred to as a Security Association (SA), between IPsec peers that is also used to manage IPsec connections, including:</p> <ul style="list-style-type: none"> • The negotiation of mutually acceptable IPsec options between peers (including peer authentication parameters, either signature based or pre-shared key based), • The establishment of additional Security Associations to protect packets flows using Encapsulating Security Payload (ESP), and • The agreement of secure bulk data encryption AES keys for use with ESP. <p>After the two peers agree upon a policy, the security parameters of the policy are identified by an SA established at each peer, and these IKE SAs apply to all subsequent IKE traffic during the negotiation.</p> <p>The TOE supports both IKEv1 and IKEv2 session establishment. As part of this support, the TOE can be configured to not support aggressive mode for IKEv1 exchanges and to only use main mode using the 'crypto isakmp aggressive-mode disable' command. The TOE supports configuration lifetimes of both Phase 1 SAs and Phase 2 SAs using the following command, lifetime. The time values for Phase 1 SAs can be limited up to 24 hours and for Phase 2 SAs up to 8 hours. The Phase 2 SA lifetimes can also be configured by an Administrator based on number of packets. The TOE supports Diffie-Hellman Group 14, 19, 20 and 24. Group 14 (2048-bit keys) can be set by using the "group 14" command in the config mode. The nonces used in IKE exchanges are generated in a manner such that the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2¹²⁸. The secret value 'x' used in the IKE Diffie-Hellman key exchange ("x" in gx mod p) is generated using a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG).</p> <p>Preshared keys can be configured using the 'crypto isakmp key' key command</p>

TOE SFRs	How the SFR is Met
	<p>and may be proposed by each of the peers negotiating the IKE establishment. The TOE supports configuring the maximum amount of traffic that is allowed to flow for a given IPsec SA using the following command, 'crypto ipsec security-association lifetime'. The default amount is 2560KB, which is the minimum configurable value. The maximum configurable value is 4GB.</p> <p>The TOE provides AES-CBC-128, AES-CBC-192 and AES-CBC-256 for encrypting the IKEv1 payloads, and AES-CBC-128, AES-CBC-192, AES-CBC-256 for IKEv2 payloads. The administrator is instructed in the AGD to ensure that the size of key used for ESP must be greater than or equal to the key size used to protect the IKE payload.</p> <p>The TOE supports Diffie-Hellman Group 14 (2048-bit keys , 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS) and 20 (384-bit Random ECP) in support of IKE Key Establishment. These keys are generated using the AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90, Table 2 in NIST SP 800-57 "Recommendation for Key Management –Part 1: General" and the following corresponding key sizes (in bits) are used: 112 (for DH Group 14), 128 (for DH Group 19), 128 (for DH Group 24), and 192 (for DH Group 20) bits. The DH group can be configured by issuing the following command during the configuration of IPsec:</p> <pre style="text-align: center;">TOE-common-criteria (config-isakmp)# group 14</pre> <p>This selects DH Group 14 (2048-bit MODP) for IKE and this sets the DH group offered during negotiations.</p> <p>The TOE generates the secret value 'x' used in the IKEv1 Diffie-Hellman key exchange ('x' in $g^x \text{ mod } p$) using the NIST approved AES-CTR Deterministic Random Bit Generator (DRBG) specified in FCS_RBG_EXT.1 and having possible lengths of 320 bits. When a random number is needed for a nonce, the probability that a specific nonce value will be repeated during the life a specific IPsec SA is less than 1 in 2^{128}. The nonce is likewise generated using the AES-CTR DRBG. This same process is used for the other supported DH groups.</p> <p>IPsec provides secure tunnels between two peers, such as two switches and remote VPN clients. An authorized administrator defines which packets are considered sensitive and should be sent through these secure tunnels. When the IPsec peer recognizes a sensitive packet, the peer sets up the appropriate secure tunnel and sends the packet through the tunnel to the remote peer. More accurately, these tunnels are sets of security associations (SAs) that are established between two IPsec peers or between the TOE and remote VPN client. The SAs define the protocols and algorithms to be applied to sensitive packets and specify the keying material to be used. SAs are unidirectional and are established per security protocol (AH or ESP). In the evaluated configuration only ESP will be configured for use.</p> <p>A crypto map (the Security Policy Definition (SPD)) set can contain multiple entries, each with a different access list. The crypto map entries are searched in a sequence - the switch attempts to match the packet to the access list (acl) specified in that entry.</p>

TOE SFRs	How the SFR is Met
	<p>When a packet matches a permit entry in a particular access list, the method of security in the corresponding crypto map is applied. If the crypto map entry is tagged as ipsec-isakmp, IPsec is triggered. Separate access lists define blocking and permitting at the interface). For example:</p> <pre>Router# access-list 101 permit ip 192.168.3.0 0.0.0.255 10.3.2.0 0.0.0.255</pre> <p>When a packet matches a permit entry in a particular access list, the method of security in the corresponding crypto map is applied. If the crypto map entry is tagged as ipsec-isakmp, IPsec is triggered. For example:</p> <pre>Router# crypto map MAP_NAME 10 ipsec-isakmp</pre> <p>The match address 101 command means to use access list 101 in order to determine which traffic is relevant. For example:</p> <pre>Router# (config-crypto-map)#match address 101</pre> <p>The traffic matching the permit acls would then flow through the IPsec tunnel and be classified as "PROTECTED".</p> <p>Traffic that does not match a permit crypto map acl and does not match a non-crypto permit acl on the interface would be DISCARDED.</p> <p>Traffic that does not match a permit acl in the crypto map, but does match a non-crypto permit acl would be allowed to BYPASS the tunnel. For example, a non-crypto permit acl for icmp would allow ping traffic to flow unencrypted if a permit crypto map was not configured that matches the ping traffic.</p> <p>The TOE implementation of the IPsec standard (in accordance with the RFCs noted in the SFR and using cryptographic algorithms AES-CBC-128, AES-CBC-192, AES-CBC-256 together with HMAC-SHA1) uses the Encapsulating Security Payload (ESP) protocol to provide authentication, encryption and anti-replay services.</p> <p>If there is no SA that the IPsec can use to protect this traffic to the peer, IPsec uses IKE to negotiate with the remote peer to set up the necessary IPsec SAs on behalf of the data flow. The negotiation uses information specified in the crypto map entry as well as the data flow information from the specific access list entry.</p> <p>In IOS the negotiations of the IKE SA adheres to configuration settings for IPsec applied by the administrator. For example in the first SA, the encryption, hash and DH group is identified, for the Child SA the encryption and the hash are identified. The administrator configures the first SA to be as strong as or stronger than the child SA; meaning if the first SA is set at AES128, then the Child SA can only be AES128. If the first SA is AES256, then the Child SA could be AES128 or AES256. During the negotiations, if a non-match is encountered, the process stops and an error message is received.</p>
FCS_SSHS_EXT.1	<p>The TOE implementation of SSHv2 supports the following:</p> <ul style="list-style-type: none"> Public key algorithms for authentication: RSA Signature Verification.

TOE SFRs	How the SFR is Met
	<ul style="list-style-type: none"> • Local password-based authentication for administrative users accessing the TOE through SSHv2, and optionally supports deferring authentication to a remote AAA server. • Encryption algorithms, AES-CBC-128, AES-CBC-256 to ensure confidentiality of the session. • The TOE's implementation of SSHv2 supports hashing algorithms hmac-sha1 and hmac-sha1-96 to ensure the integrity of the session. • The TOE's implementation of SSHv2 can be configured to only allow Diffie-Hellman Group 14 (2048-bit keys) Key Establishment, as required by the PP. • Packets greater than 65,535 bytes in an SSH transport connection are dropped. Large packets are detected by the SSH implementation, and dropped internal to the SSH process. • The TOE can also be configured to ensure that SSH re-key of no longer than one hour and no more than one gigabyte of transmitted data for the session key.
FCS_RBG_EXT.1	<p>The TOE implements a NIST-approved AES-CTR Deterministic Random Bit Generator (DRBG), as specified in SP 800-90 seeded by an entropy source that accumulates entropy from a TSF-hardware based noise source.</p> <p>The deterministic RBG is seeded with a minimum of 256 bits of entropy, which is at least equal to the greatest security strength of the keys and hashes that it will generate.</p>
FIA_AFL.1	<p>The TOE provides the privileged administrator the ability to specify the maximum number of unsuccessful authentication attempts before privileged administrator or non-privileged administrator is locked out through the administrative CLI using a privileged CLI command. While the TOE supports a range from 1-25, in the evaluated configuration, the maximum number of failed attempts is recommended to be set to 3.</p> <p>When a privileged administrator or non-privileged administrator attempting to log into the administrative CLI reaches the administratively set maximum number of failed authentication attempts, the user will not be granted access to the administrative functionality of the TOE until a privileged administrator resets the user's number of failed login attempts through the administrative CLI.</p>
FIA_PMG_EXT.1	<p>The TOE supports the local definition of users with corresponding passwords. The passwords can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "\$", "%", "^", "&", "*", "(, ", "). Minimum password length is settable by the Authorized Administrator, and can be configured for minimum password lengths of 15 characters.</p>
FIA_UIA_EXT.1	<p>The TOE requires all users to be successfully identified and authenticated before allowing any TSF mediated actions to be performed except for the login warning banner that is displayed prior to user authentication and any network packets as configured by the authorized administrator may flow through the switch.</p>
FIA_UAU_EXT.2	

TOE SFRs	How the SFR is Met
	<p>Administrative access to the TOE is facilitated through the TOE's CLI. The TOE mediates all administrative actions through the CLI. Once a potential administrative user attempts to access the CLI of the TOE through either a directly connected console or remotely through an SSHv2 secured connection, the TOE prompts the user for a user name and password. Only after the administrative user presents the correct authentication credentials will access to the TOE administrative functionality be granted. No access is allowed to the administrative functionality of the TOE until an administrator is successfully identified and authenticated.</p> <p>The TOE provides a local password based authentication mechanism as well as RADIUS AAA server for remote authentication.</p> <p>The administrator authentication policies include authentication to the local user database or redirection to a remote authentication server. Interfaces can be configured to try one or more remote authentication servers, and then fail back to the local user database if the remote authentication servers are inaccessible.</p> <p>The process for authentication is the same for administrative access whether administration is occurring via a directly connected console or remotely via SSHv2 secured connection.</p> <p>At initial login, the administrative user is prompted to provide a username. After the user provides the username, the user is prompted to provide the administrative password associated with the user account. The TOE then either grant administrative access (if the combination of username and password is correct) or indicate that the login was unsuccessful. The TOE does not provide a reason for failure in the cases of a login failure.</p>
FIA_UAU.7	<p>When a user enters their password at the local console, the TOE displays only '*' characters so that the user password is obscured. For remote session authentication, the TOE does not echo any characters as they are entered. The TOE does not provide a reason for failure in the cases of a login failure.</p>
FIA_X509_EXT.1/Rev	<p>The TOE uses X.509v3 certificates as defined by RFC 5280 to support authentication for IPsec connections.</p>
FIA_X509_EXT.2	<p>Public key infrastructure (PKI) credentials, such as Rivest, Shamir, and Adelman (RSA) and Elliptical Curve Digital Signature Algorithm (ECDSA) keys and certificates can be stored in a specific location on the router, such as NVRAM and flash memory. The certificates themselves provide protection in that they are digitally signed. If a certificate is modified in any way, it would be invalidated. Only authorized administrators with the necessary privilege level can access the certificate storage and add/delete them. The digital signature verifications process would show that the certificate had been tampered with when the hash value would be invalid. The physical security of the router (A.Physical) protects the router and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. CRL are configurable and may be used for certificate revocation. Checking is also done for the basicConstraints extension and the cA flag to determine whether they are present and set to TRUE. If they are not, the certificate is not accepted.</p>
FIA_X509_EXT.3	<p>The physical security of the router (A.Physical) protects the router and the certificates from being tampered with or deleted. In addition, the TOE identification and authentication security functions protect an unauthorized user from gaining access to the TOE. CRL are configurable and may be used for certificate revocation. Checking is also done for the basicConstraints extension and the cA flag to determine whether they are present and set to TRUE. If they are not, the certificate is not accepted.</p>

TOE SFRs	How the SFR is Met
	<p>If the TOE does not have the applicable CRL and is unable to obtain one, the administrator is able to choose whether or not to accept the certificate.</p> <p>The certificate chain path validation is configured on the TOE by first setting crypto pki trustpoint name and then configuring the level to which a certificate chain is processed on all certificates including subordinate CA certificates using the <i>chain-validation</i> command.</p>
FMT_MOF.1/ManualUpdate	<p>The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the privileged and semi-privileged levels. For the purposes of this evaluation, the privileged level is equivalent to full administrative access to the CLI, which is the default access for IOS privilege level 15; and the semi-privileged level equates to any privilege level that has a subset of the privileges assigned to level 15. Privilege levels 0 and 1 are defined by default and are customizable, while levels 2-14 are undefined by default and also customizable.</p>
FMT_MTD.1/CoreData	<p>The term “Authorized Administrator” is used in this ST to refer to any user that has been assigned to a privilege level that is permitted to perform the relevant action; therefore, has the appropriate privileges to perform the requested functions. As such the semi-privileged administrators with only a subset of privileges may also manage and modify TOE data based on the privileges assigned.</p> <p>The TOE provides the ability for Security Administrators (a.k.a Authorized Administrators) to access TOE data, such as audit data, configuration data, security attributes, session thresholds, cryptographic keys and updates. Each of the predefined and administratively configured privilege level has a set of permissions that will grant them access to the TOE data, though with some privilege levels, the access is limited.</p> <p>The TOE does not provide automatic updates to the software version running on the TOE.</p> <p>The Security Administrators (a.k.a Authorized Administrators) can query the software version running on the TOE, and can initiate updates to (replacements of) software images. When software updates are made available by Cisco, the Authorized Administrators can obtain, verify the integrity of, and install those updates.</p> <p>In addition, the warning and access banner can be displayed prior to the identification and authentication of an authorized administrator. However, no administrative functionality is available prior to administrative login.</p>
FMT_SMF.1	<p>The TOE provides all the capabilities necessary to securely manage the TOE and the services provided by the TOE. The management functionality of the TOE is provided through the TOE CLI. The specific management capabilities available from the TOE include -</p> <ul style="list-style-type: none"> • Local and remote administration of the TOE and the services provided by the TOE via the TOE CLI, as described above; • The ability to manage the warning banner message and content – allows the Authorized Administrator the ability to define warning banner that is displayed prior to establishing a session (note this applies to the interactive (human) users; e.g. administrative users

TOE SFRs	How the SFR is Met
	<ul style="list-style-type: none"> • The ability to manage the time limits of session inactivity which allows the Authorized Administrator the ability to set and modify the inactivity time threshold. • The ability to update the IOS-XE software. The validity of the image is provided using both SHA-512 and digital signature prior to installing the update • The ability to manage audit behavior and the audit logs which allows the Authorized Administrator to configure the audit logs, view the audit logs, and to clear the audit logs • The ability to display the log on banner and to allow any network packets as configured by the authorized administrator may flow through the switch prior to the identification and authentication process • The ability to manage the cryptographic functionality which allows the Authorized Administrator the ability to identify and configure the algorithms used to provide protection of the data, such as generating the RSA keys to enable SSHv2 • Ability to configure the IPsec functionality. • Ability to import the X.509v3 certificates. <p>Information about TSF-initiated Termination is covered in the TSS under FTA_SSL_EXT.1 or FTA_SSL.3.</p>
FMT_SMR.2	<p>The TOE platform maintains privileged and semi-privileged administrator roles. The TOE performs role-based authorization, using TOE platform authorization mechanisms, to grant access to the semi-privileged and privileged roles. For the purposes of this evaluation, the privileged role is equivalent to full administrative access to the CLI, which is the default access for IOS-XE privilege level 15; and the semi-privileged role equates to any privilege level that has a subset of the privileges assigned to level 15. Privilege levels 0 and 1 are defined by default and are customizable, while levels 2-14 are undefined by default and are also customizable. Note: the levels are not hierarchical.</p> <p>The term “Security Administrator” is used in this ST to refer to any user which has been assigned to a privilege level that is permitted to perform the relevant action; therefore has the appropriate privileges to perform the requested functions.</p> <p>The privilege level determines the functions the user can perform; hence the Security Administrator with the appropriate privileges. Refer to the Guidance documentation and IOS-XE Command Reference Guide for available commands and associated roles and privilege levels.</p> <p>The TOE can and shall be configured to authenticate all access to the command line interface using a username and password.</p> <p>The TOE supports both local administration via a directly connected console cable and remote administration via SSH or IPsec over SSH.</p>

TOE SFRs	How the SFR is Met
FPT_SKP_EXT.1 FPT_APW_EXT.1	<p>The TOE stores all private keys in a secure directory protected from access as there is no interface in which the keys can be accessed.</p> <p>The TOE includes CLI command features that can be used to configure the TOE to encrypt all locally defined user passwords. In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators. The password is encrypted by using the command "password encryption aes" used in global configuration mode.</p> <p>The command <i>service password-encryption</i> applies encryption to all passwords, including username passwords, authentication key passwords, the privileged command password, console and virtual terminal line access passwords.</p> <p>Additionally, enabling the 'hidekeys' command in the logging configuration ensures that passwords are not displayed in plaintext.</p> <p>The TOE includes a Master Passphrase feature that can be used to configure the TOE to encrypt all locally defined user passwords using AES. In this manner, the TOE ensures that plaintext user passwords will not be disclosed even to administrators.</p> <p>Password encryption is configured using the 'service password-encryption' command. There are no administrative interfaces available that allow passwords to be viewed as they are encrypted via the password-encryption service.</p>
FPT_STM_EXT.1	<p>The TOE provides a source of date and time information used in audit event timestamps, and for certificate validity checking. The clock function is reliant on the system clock provided by the underlying hardware. This date and time is used as the time stamp that is applied to TOE generated audit records and used to track inactivity of administrative sessions. The time information is also used in various routing protocols such as, OSPF, BGP, and ERF; Set system time, Calculate IKE stats (including limiting SAs based on times); determining AAA timeout, and administrative session timeout.</p>
FPT_TUD_EXT.1	<p>An Authorized Administrator can query the software version running on the TOE and can initiate updates to software images. The current active version can be verified by executing the "show version" command from the TOE's CLI. When software updates are made available by Cisco, an administrator can obtain, verify the integrity of, and install those updates. The updates can be downloaded from the software.cisco.com.</p> <p>The cryptographic hashes (i.e., SHA-512) are used to verify software update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to actually update the applicable TOE components. Authorized Administrators can download the approved image file from Cisco.com onto a trusted computer system for usage in the trusted update functionality. The hash value can be displayed by hovering over the software image name under details on the Cisco.com web site. The verification should not be performed on the TOE during the update process. If the hashes do not match, contact Cisco Technical Assistance Center (TAC).</p>

TOE SFRs	How the SFR is Met
	<p>Digital signatures and published hash mechanisms are used to verify software/firmware update files (to ensure they have not been modified from the originals distributed by Cisco) before they are used to actually update the applicable TOE components. The TOE image files are digitally signed so their integrity can be verified during the boot process, and an image that fails an integrity check will not be loaded.</p> <p>To verify the digital signature prior to installation, the “show software authenticity file” command allows you to display software authentication related information that includes image credential information, key type used for verification, signing information, and other attributes in the signature envelope, for a specific image file. If the output from the “show software authenticity file” command does not provide the expected output, contact Cisco Technical Assistance Center (TAC) https://tools.cisco.com/ServiceRequestTool/create/launch.do.</p> <p>Further instructions for how to do this verification are provided in the administrator guidance for this evaluation.</p> <p>Software images are available from Cisco.com at the following: http://www.cisco.com/cisco/software/navigator.html</p>
FPT_TST_EXT.1	<p>The TOE runs a suite of self-tests during initial start-up to verify its correct operation. Refer to the FIPS Security Policy for available options and management of the cryptographic self-test. For testing of the TSF, the TOE automatically runs checks and tests at startup, during resets and periodically during normal operation to ensure the TOE is operating correctly, including checks of image integrity and all cryptographic functionality.</p> <p>During the system bootup process (power on or reboot), all the Power on Startup Test (POST) components for all the cryptographic modules perform the POST for the corresponding component (hardware or software). These tests include:</p> <ul style="list-style-type: none"> • AES Known Answer Test – For the encrypt test, a known key is used to encrypt a known plain text value resulting in an encrypted value. This encrypted value is compared to a known encrypted value to ensure that the encrypt operation is working correctly. The decrypt test is just the opposite. In this test a known key is used to decrypt a known encrypted value. The resulting plaintext value is compared to a known plaintext value to ensure that the decrypt operation is working correctly. • RSA Signature Known Answer Test (both signature/verification) – This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly. • RNG/DRBG Known Answer Test – For this test, known seed values are provided to the DRBG implementation. The DRBG uses these values to generate random bits. These random bits are

TOE SFRs	How the SFR is Met
	<p>compared to known random bits to ensure that the DRBG is operating correctly.</p> <ul style="list-style-type: none"> • HMAC Known Answer Test – For each of the hash values listed, the HMAC implementation is fed known plaintext data and a known key. These values are used to generate a MAC. This MAC is compared to a known MAC to verify that the HMAC and hash operations are operating correctly. • Software Integrity Test – The Software Integrity Test is run automatically whenever the IOS system images is loaded and confirms that the image file that's about to be loaded has maintained its integrity. • SHA-1/256/512 Known Answer Test – For each of the values listed, the SHA implementation is fed known data and key. These values are used to generate a hash. This hash is compared to a known value to verify they match and the hash operations are operating correctly. • ECDSA self-test – This test takes a known plaintext value and Private/Public key pair and used the public key to encrypt the data. This value is compared to a known encrypted value to verify that encrypt operation is working properly. The encrypted data is then decrypted using the private key. This value is compared to the original plaintext value to ensure the decrypt operation is working properly. <p>If any component reports failure for the POST, the system crashes and appropriate information is displayed on the screen, and saved in the crashinfo file.</p> <p>All ports are blocked from moving to forwarding state during the POST. If all components of all modules pass the POST, the system is placed in FIPS PASS state and ports are allowed to forward data traffic.</p> <p>If an error occurs during the self-test, a SELF_TEST_FAILURE system log is generated.</p> <p>Example Error Message <code>_FIPS-2-SELF_TEST_IOS_FAILURE: "IOS crypto FIPS self test failed at %s."</code> Explanation FIPS self test on IOS crypto routine failed.</p> <p>These tests are sufficient to verify that the correct version of the TOE software is running as well as that the cryptographic operations are all performing as expected because any deviation in the TSF behavior will be identified by the failure of a self-test.</p> <p>The integrity of stored TSF executable code when it is loaded for execution can be verified through the use of RSA and Elliptic Curve Digital Signature algorithms.</p>

TOE SFRs	How the SFR is Met
FTA_SSL_EXT.1	An administrator can configure maximum inactivity times individually for both local and remote administrative sessions through the use of the “session-timeout” setting applied to the console. When a session is inactive (i.e., no session input from the administrator) for the configured period of time the TOE will terminate the session, and no further activity is allowed requiring the administrator to log in (be successfully identified and authenticated) again to establish a new session. If a remote user session is inactive for a configured period of time, the session will be terminated and will require authentication to establish a new session.
FTA_SSL.3	The allowable inactivity timeout range is from 1 to 65535 seconds. Administratively configurable timeouts are also available for the EXEC level access (access above level 1) through use of the “exec-timeout” setting.
FTA_SSL.4	An administrator is able to exit out of both local and remote administrative sessions. Each administrator logged onto the TOE can manually terminate their session using the “exit” or “logout” command.
FTA_TAB.1	The TOE displays a privileged Administrator specified banner on the CLI management interface prior to allowing any administrative access to the TOE. This interface is applicable for both local (via console) and remote (via SSH) TOE administration.
FTP_ITC.1	<p>The TOE protects communications with peer or neighbor routers using keyed hash as defined in FCS_COP.1/KeyedHash and cryptographic hashing functions FCS_COP.1/Hash. This protects the data from modification of data by hashing that verify that data has not been modified in transit. In addition, encryption of the data as defined in FCS_COP.1/DataEncryption is provided to ensure the data is not disclosed in transit.</p> <p>The TOE protects communications between the TOE and the remote audit server using IPsec. This provides a secure channel to transmit the log events. Likewise communications between the TOE and AAA servers are secured using IPsec.</p>
FTP_TRP.1 /Admin	All remote administrative communications take place over a secure encrypted SSHv2 session. The SSHv2 session is encrypted using AES encryption. The remote users are able to initiate SSHv2 communications with the TOE.

7 ANNEX A: KEY ZEROIZATION

7.1 Key Zeroization

The following table describes the key zeroization referenced by FCS_CKM.4 provided by the TOE.

Table 20 TOE Key Zeroization

Name	Description	Zeroization
Diffie-Hellman Shared Secret	This is the shared secret used as part of the Diffie-Hellman key exchange. This key is stored in SDRAM.	Automatically after completion of DH exchange. Overwritten with: 0x00
Diffie Hellman private exponent	This is the private exponent used as part of the Diffie-Hellman key exchange. This key is stored in SDRAM.	Zeroized upon completion of DH exchange. Overwritten with: 0x00
skeyid	This is an IKE intermittent value used to create skeyid_d. This information is stored in DRAM. This information and keys are stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
skeyid_d	This is an IKE intermittent value used to derive keying data for IPsec. This information and keys are stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session encrypt key	This the key IPsec key used for encrypting the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
IKE session authentication key	This the key IPsec key used for authenticating the traffic in an IPsec connection. This key is stored in SDRAM.	Automatically after IKE session terminated. Overwritten with: 0x00
ISAKMP preshared	This is the configured pre-shared key for ISAKMP negotiation. This key is stored in SDRAM.	Zeroized using the following command: # no crypto isakmp key Overwritten with: 0x0d
IKE ECDSA Private Key	The ECDSA private-public key pair is created by the device itself using the key generation CLI command. Afterwards, the device's public key must be put into the device certificate. The device's certificate is created by creating a trustpoint on the device. This trustpoint authenticates with the	Zeroized using the following command: # crypto key zeroize ecdsa ¹

¹ Issuing this command will zeroize/delete all ECDSA keys on the TOE.

Name	Description	Zeroization
	<p>CA server to get the CA certificate and also enrolls with the CA server to generate the device certificate.</p> <p>In the IKE authentication step, the device's certificate is firstly sent to other device to be authenticated. The other device verifies that the certificate is signed by CA's signing key, then sends back a random secret encrypted by the device's public key in the valid device certificate. Only the device with the matching device private key can decrypt the message and obtain the random secret. This key is stored in NVRAM.</p>	Overwritten with: 0x0d
IKE RSA Private Key	<p>The RSA private-public key pair is created by the device itself using the key generation CLI described below.</p> <p>The device's public key must be added into the device certificate. The device's certificate is created by creating a trustpoint on the device. This trustpoint authenticates with the CA server to get the CA certificate and to enrol with the CA server to generate the device certificate.</p> <p>In the IKE authentication step, the device's certificate is first sent to other device so that it can be authenticated. The other device verifies the certificate is signed by CA's signing key, and then the device sends a random secret encrypted by the device's public key in the valid device certificate. Thus establishing the trusted connection since only the device with the matching device private key can decrypt the message and obtain the random secret. This key is stored in NVRAM.</p>	<p>Zeroized using the following command:</p> <pre># crypto key zeroize rsa</pre> <p>Overwritten with: 0x0d</p>
IPsec encryption key	This is the key used to encrypt IPsec sessions. This key is stored in SDRAM.	<p>Automatically when IPsec session terminated.</p> <p>Overwritten with: 0x00</p>
IPsec authentication key	This is the key used to authenticate IPsec sessions. This key is stored in SDRAM.	<p>Automatically when IPsec session terminated.</p> <p>Overwritten with: 0x00</p>
RADIUS secret	Shared secret used as part of the Radius authentication method. The password is stored in NVRAM.	<p>Zeroized using the following command:</p> <pre># no radius-server key</pre> <p>Overwritten with: 0x0d</p>
SSH Private Key	This is the private (secret) key of the asymmetric key pair required to establish the secure SSH session. The key is stored in NVRAM.	<p>Zeroized using the following command:</p> <pre># crypto key zeroize rsa</pre> <p>Overwritten with: 0x00</p>
SSH Session Key	Once the function has completed the operations requiring the RSA key object, the module over writes the entire object (no matter its contents). This key is stored in SDRAM.	Automatically when the SSH session is terminated.

Name	Description	Zeroization
		Overwritten with: 0x00
User Password	This is a variable 15+-character password that is used to authenticate local users. The password is stored in NVRAM.	Zeroized by overwriting with new password
Enable Password (if used)	This is a variable 15+-character password that is used to authenticate local users at a higher privilege level. The password is stored in NVRAM.	Zeroized by overwriting with new password
RNG Seed	This seed is for the RNG. The seed is stored in SDRAM.	Zeroized upon power cycle the device
RNG Seed Key	This is the seed key for the RNG. The seed key is stored in SDRAM.	Zeroized upon power cycle the device

8 ANNEX B: REFERENCES

The following documentation was used to prepare this ST:

Table 21 References

Identifier	Description
[CC_PART1]	Common Criteria for Information Technology Security Evaluation – Part 1: Introduction and general model, dated September 2012, version 3.1, Revision 4
[CC_PART2]	Common Criteria for Information Technology Security Evaluation – Part 2: Security functional components, dated September 2012, version 3.1, Revision 4
[CC_PART3]	Common Criteria for Information Technology Security Evaluation – Part 3: Security assurance components, September 2012, version 3.1, Revision 4
[CEM]	Common Methodology for Information Technology Security Evaluation – Evaluation Methodology, September 2012, version 3.1, Revision 4
[NDcPP]	collaborative Protection Profile for Network Devices, + Errata 20180314, Version 2.0e, 14 March 2018
[800-56A]	NIST Special Publication 800-56A, March, 2007 Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)
[800-56B]	NIST Special Publication 800-56B Recommendation for Pair-Wise, August 2009 Key Establishment Schemes Using Integer Factorization Cryptography
[FIPS 140-2]	FIPS PUB 140-2 Federal Information Processing Standards Publication Security Requirements for Cryptographic Modules May 25, 2001
[FIPS PUB 186-3]	FIPS PUB 186-3 Federal Information Processing Standards Publication Digital Signature Standard (DSS) June, 2009
[FIPS PUB 186-4]	FIPS PUB 186-4 Federal Information Processing Standards Publication Digital Signature Standard (DSS) July 2013
[FIPS PUB 198-1]	Federal Information Processing Standards Publication The Keyed-Hash Message Authentication Code (HMAC) July 2008
[NIST SP 800-90A Rev 1]	NIST Special Publication 800-90A Recommendation for Random Number Generation Using Deterministic Random Bit Generators January 2015
[FIPS PUB 180-3]	FIPS PUB 180-3 Federal Information Processing Standards Publication Secure Hash Standard (SHS) October 2008